

(19) 世界知的所有権機関  
国際事務局(43) 国際公開日  
2002年10月10日 (10.10.2002)

PCT

(10) 国際公開番号  
WO 02/080446 A1

(51) 国際特許分類: H04L 9/08, G06F 17/60  
 (72) 発明者: および  
 (75) 発明者/出願人 (米国についてのみ): 石黒 隆二 (ISHIGURO, Ryuji) [JP/JP]; 〒141-0001 東京都 品川区 北品川 6丁目 7番 35号 ソニー株式会社内 Tokyo (JP).

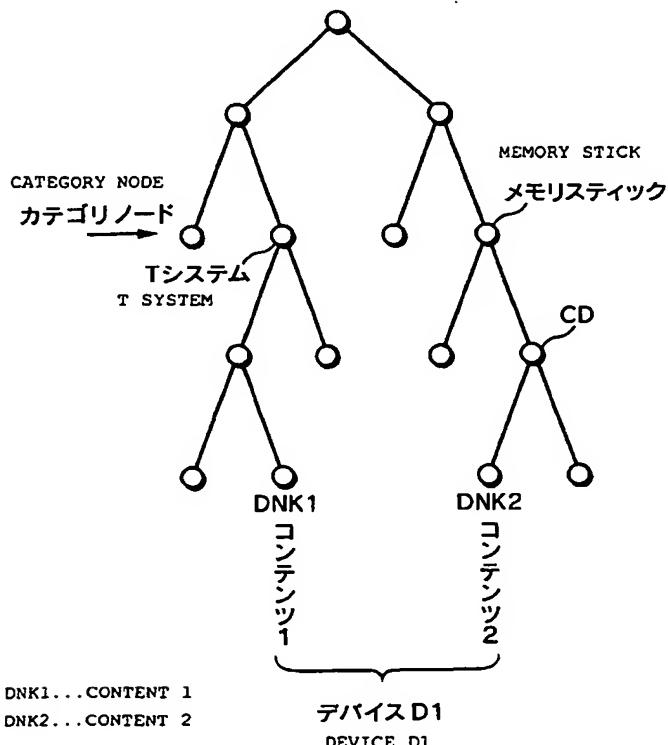
(21) 国際出願番号: PCT/JP02/02955  
 (22) 国際出願日: 2002年3月27日 (27.03.2002)  
 (25) 国際出願の言語: 日本語  
 (26) 国際公開の言語: 日本語  
 (30) 優先権データ:  
 特願2001-94803 2001年3月29日 (29.03.2001) JP  
 (71) 出願人 (米国を除く全ての指定国について): ソニー株式会社 (SONY CORPORATION) [JP/JP]; 〒141-0001 東京都 品川区 北品川 6丁目 7番 35号 Tokyo (JP).

(74) 代理人: 稲本 義雄 (INAMOTO, Yoshio); 〒160-0023 東京都 新宿区 西新宿 7丁目 11番 18号 711ビルディング 4階 Tokyo (JP).  
 (81) 指定国 (国内): CN, JP, KR, US.  
 (84) 指定国 (広域): ヨーロッパ特許 (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR).  
 添付公開書類:  
 — 國際調査報告書

/統葉有/

(54) Title: INFORMATION PROCESSING APPARATUS

(54) 発明の名称: 情報処理装置



(57) Abstract: A license server providing a key for decoding a content allocates a leaf of a key management hierarchical tree structure to a client, generates a set of node keys as a device node key and transmits it together with a leaf ID, a client secret key, and the like to the client. By making an intermediate node in the hierarchical structure a summit and making nodes below the summit node the related nodes of the category defined by the summit node, a manufacturer and a content provider managing the summit node of the category can generate a validation key block with that node as a summit and deliver it to the devices below the summit node. Thus, it is possible to update a key without affecting devices belonging to other categories.

WO 02/080446 A1

/統葉有/

BEST AVAILABLE COPY



2文字コード及び他の略語については、定期発行される各PCT gazetteの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

---

(57) 要約:

コンテンツを復号するための鍵を提供するライセンスサーバは、鍵管理階層ツリー構造のリーフをクライアントに割り当て、ノードキーの組をデバイスノードキーとして生成し、リーフID、クライアントの秘密鍵等と共にクライアントに送信する。

鍵階層構造の中間のノードを頂点として、以下のノードをその頂点ノードに定義されたカテゴリの関連ノードとして設定することにより、カテゴリの頂点ノードを管理するメーカー、コンテンツプロバイダ等がそのノードを頂点とする有効化キーブロックを独自に生成して頂点ノード以下に属するデバイスに配信することが可能となり、他のカテゴリに属するデバイスに影響を及ぼすことなくキーを更新することができる。

## 明細書

## 情報処理装置

## 技術分野

5 本発明は、情報処理装置に関し、特に、異なるカテゴリに属するコンテンツを  
1 つのデバイスで管理することができるようとした情報処理装置に関する。

## 背景技術

最近、インターネットが普及し、インターネットを介してオーディオやビデオ  
10 などの各種のコンテンツを配信することが提案され、一部、実際に実現されてい  
る。この場合、コンテンツの著作権を保護するためコンテンツには様々な利用条  
件が定められ、デバイスはその利用条件を満たす利用しかできないようになって  
いる。

しかしながら、従来、1 つのデバイスにおいて、全てのコンテンツに対して同  
15 様の管理形態で管理が行われていた。その結果、1 つのデバイスで、サービス、  
利用条件、コンテンツの供与元などが異なるカテゴリごとにコンテンツの管理形  
態を変えることができず、コンテンツの提供者にとっては柔軟な管理を行うこと  
ができず、デバイスの操作性にも問題があった。

## 20 発明の開示

本発明は、このような状況に鑑みてなされたものであり、サービス、利用条件、  
コンテンツの提供元など異なるカテゴリに属する複数のコンテンツを1 つのデ  
バイスで利用することができるようになり、コンテンツ提供者、利用者の利便、デ  
バイスの操作性を改善するものである。

25 本発明の第1 の情報処理装置は、他の情報処理装置を、所定の階層の第1 のノ  
ードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造にお  
ける、第1 のノードの下位にあるリーフに、一意に割り当てる割り当て手段と、

割り当て手段により割り当てられたリーフから第1のノードまでのパスに対応するデバイスノードキーを、他の情報処理装置に提供する提供手段とを備えることを特徴とする。

前記デバイスノードキーと共にリーフを識別するリーフ識別情報を提供するよ  
5 うにすることができる。

前記他の情報処理装置からリーフ識別情報とコンテンツの利用を許可するためのライセンスを識別するライセンス識別情報とを含むライセンス要求を受信する受信手段と、ライセンス要求に含まれるリーフ識別情報が付加され電子署名がなされたライセンスを送信する送信手段とをさらに備えるようにすることができる。

10 前記デバイスノードキーと共に、他の情報処理装置の秘密鍵および公開鍵、並びに自分自身の公開鍵を提供するようにすることができる。

本発明の第1の情報処理方法は、他の情報処理装置を、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフに、一意に割り当てる割り当てステップと、割り当てステップの処理により割り当てられたリーフから第1のノードまでのパスに対応するデバイスノードキーを、他の情報処理装置に提供する提供ステップとを含むことを特徴とする。

本発明の第1の記録媒体のプログラムは、他の情報処理装置を、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフに、一意に割り当てる割り当てステップと、割り当てステップの処理により割り当てられたリーフから第1のノードまでのパスに対応するデバイスノードキーを、他の情報処理装置に提供する提供ステップとを含むことを特徴とする。

本発明の第1のプログラムは、他の情報処理装置を、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフに、一意に割り当てる割り当てステップと、割り当てステップの処理により割り当てられたリーフから第1のノードまでのパ

スに対応するデバイスノードキーを、他の情報処理装置に提供する提供ステップとをコンピュータに実現させる。

本発明の第2の情報処理装置は、他の情報処理装置から供給された、他の情報処理装置に割り当てられた鍵管理階層ツリー構造のリーフを識別するリーフ識別

5 情報を受信する受信手段と、受信手段により受信されたリーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフのリーフ識別情報である場合に、リーフ識別情報を含むライセンスを送信する送信手段とを備えることを特徴とする。

10 本発明の第2の情報処理方法は、他の情報処理装置から供給された、他の情報処理装置に割り当てられた鍵管理課位相ツリー構造のリーフを識別するリーフ識別情報を受信する受信ステップと、受信ステップの処理により受信されたリーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフのリ  
15 ーフ識別情報である場合に、リーフ識別情報を含むライセンスを送信する送信ステップとを含むことを特徴とする。

本発明の第2の記録媒体のプログラムは、他の情報処理装置から供給された、他の情報処理装置に割り当てられた鍵管理課位相ツリー構造のリーフを識別するリーフ識別情報を受信する受信ステップと、受信ステップの処理により受信され  
20 たリーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフのリーフ識別情報である場合に、リーフ識別情報を含むライセンスを送信する送信ステップとを含むことを特徴とする。

本発明の第2のプログラムは、他の情報処理装置から供給された、他の情報処理装置に割り当てられた鍵管理課位相ツリー構造のリーフを識別するリーフ識別情報を受信する受信ステップと、受信ステップの処理により受信されたリーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当て

られた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフのリーフ識別情報である場合に、リーフ識別情報を含むライセンスを送信する送信ステップとをコンピュータに実現させる。

本発明の第3の情報処理装置は、コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求を受信する受信手段と、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第1のノードの下位にあるリーフから第1のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック（EKB）を含む暗号化コンテンツを送信する送信手段とを備えることを特徴とする。

10 本発明の第3の情報処理方法は、コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求を受信する受信手段と、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第1のノードの下位にあるリーフから第1のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック（EKB）を含む暗号化コンテンツを送信する送信手段とを含むことを特徴とする。

本発明の第3の記録媒体のプログラムは、コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求を受信する受信手段と、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第1のノードの下位にあるリーフから第1のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック（EKB）を含む暗号化コンテンツを送信する送信手段とを含むことを特徴とする。

本発明の第3のプログラムは、コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求を受信する受信手段と、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第1のノードの下位にあるリーフから第1のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック（EKB）を含む暗号化コンテンツを送信する送信手段とをコンピュータに実現させる。

本発明の第4の情報処理装置は、自分自身に割り当てられた、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第1のノードの下位にあるリーフから、第1のノードまでに対応するデバイスノードキーを記憶する記憶手段と、第1のノードをルートキーに対応させた有効化キーブロック (EKB) を含む暗号化コンテンツを取得するコンテンツ取得手段と、コンテンツ取得手段により取得された暗号化コンテンツに含まれる有効化キーブロック (EKB) を記憶手段に記憶されているデバイスノードキーを用いて復号処理することで、暗号化コンテンツを復号する復号手段と、復号手段により復号されたコンテンツを出力する出力手段とを備えることを特徴とする。

本発明の第4の情報処理方法は、自分自身に割り当てられた、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第1のノードの下位にあるリーフから、第1のノードまでに対応するデバイスノードキーの記憶を制御する記憶制御ステップと、第1のノードをルートキーに対応させた有効化キーブロック (EKB) を含む暗号化コンテンツを取得するコンテンツ取得ステップと、コンテンツ取得ステップの処理により取得されたコンテンツに含まれる有効化キーブロック (EKB) を記憶制御ステップの処理により記憶が制御されているデバイスノードキーを用いて復号処理することで、暗号化コンテンツを復号する復号ステップと、復号ステップの処理により復号されたコンテンツを出力する出力ステップとを含むことを特徴とする。

本発明の第4の記録媒体のプログラムは、自分自身に割り当てられた、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第1のノードの下位にあるリーフから、第1のノードまでに対応するデバイスノードキーの記憶を制御する記憶制御ステップと、第1のノードをルートキーに対応させた有効化キーブロック (EKB) を含む暗号化コンテンツを取得するコンテンツ取得ステップと、コンテンツ取得ステップの処理により取得されたコンテンツに含まれる有効化キーブロック (EKB) を記憶制御ス

テップの処理により記憶が制御されているデバイスノードキーを用いて復号処理することで、暗号化コンテンツを復号する復号ステップと、復号ステップの処理により復号されたコンテンツを出力する出力ステップとを含むことを特徴とする。

本発明の第4のプログラムは、自分自身に割り当てられた、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第1のノードの下位にあるリーフから、第1のノードまでに対応するデバイスノードキーの記憶を制御する記憶制御ステップと、第1のノードをルートキーに対応させた有効化キーブロック（EKB）を含む暗号化コンテンツを取得するコンテンツ取得ステップと、コンテンツ取得ステップの処理により取得されたコンテンツに含まれる有効化キーブロック（EKB）を記憶制御ステップの処理により記憶が制御されているデバイスノードキーを用いて復号処理することで、暗号化コンテンツを復号する復号ステップと、復号ステップの処理により復号されたコンテンツを出力する出力ステップとをコンピュータに実現させる。

本発明の第1の情報処理装置および方法、並びに、プログラムにおいては、他の情報処理装置が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフに、一意に割り当てられ、その割り当てられたリーフから第1のノードまでのパスに対応するデバイスノードキーが、他の情報処理装置に提供される。

本発明の第2の情報処理装置および方法、並びに、プログラムにおいては、他の情報処理装置から供給された、他の情報処理装置に割り当てられた鍵管理階層ツリー構造のリーフを識別するリーフ識別情報が受信され、その受信されたリーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、第1のノードの下位にあるリーフのリーフ識別情報である場合に、リーフ識別情報を含むライセンスが送信される。

本発明の第3の情報処理装置および方法、並びに、プログラムにおいては、コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求が受信され、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階

層ツリー構造における第 1 のノードの下位にあるリーフから第 1 のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック (EKB) を含む暗号化コンテンツが送信される。

本発明の第 4 の情報処理装置および方法、並びに、プログラムにおいては、自分自身に割り当てられた、所定の階層の第 1 のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における第 1 のノードの下位にあるリーフから、第 1 のノードまでに対応するデバイスノードキーが記憶され、第 1 のノードをルートキーに対応させた有効化キーブロック (EKB) を含む暗号化コンテンツが取得され、その取得された暗号化コンテンツに含まれる有効化キー 5 ブロック (EKB) がデバイスノードキーを用いて復号処理されることで、暗号化コンテンツが復号され、その復号されたコンテンツが出力される。

10

#### 図面の簡単な説明

図 1 は、本発明を適用したコンテンツ提供システムの構成を示すブロック図で 15 ある。

図 2 は、図 1 のクライアントの構成を示すブロック図である。

図 3 は、図 1 のクライアントのコンテンツのダウンロード処理を説明するフローチャートである。

図 4 は、図 1 のコンテンツサーバのコンテンツ提供処理を説明するフローチャ 20 ートである。

図 5 は、図 4 のステップ S 2 6 におけるフォーマットの例を示す図である。

図 6 は、図 1 のクライアントのコンテンツ再生処理を説明するフローチャートである。

図 7 は、図 6 のステップ S 4 3 のライセンス取得処理の詳細を説明するフロ 25 チャートである。

図 8 は、ライセンスの構成を示す図である。

図9は、図1のライセンスサーバのライセンス提供の処理を説明するフローチャートである。

図10は、図6のステップS45におけるライセンス更新処理の詳細を説明するフローチャートである。

5 図11は、図1のライセンスサーバのライセンス更新処理を説明するフローチャートである。

図12は、キーの構成を説明する図である。

図13は、カテゴリノードを説明する図である。

図14は、ノードとデバイスの対応の具体例を示す図である。

10 図15Aは、有効化キーブロックの構成を説明する図である。

図15Bは、有効化キーブロックの構成を説明する図である。

図16は、有効化キーブロックの利用を説明する図である。

図17は、有効化キーブロックのフォーマットの例を示す図である。

図18は、有効化キーブロックのタグの構成を説明する図である。

15 図19は、DNKを用いたコンテンツの復号処理を説明する図である。

図20は、有効化キーブロックの例を示す図である。

図21は、複数のコンテンツの1つのデバイスに対する割り当てを説明する図である。

図22は、ライセンスのカテゴリを説明する図である。

20 図23は、登録処理を説明するタイミングチャートである。

図24は、クライアントのリッピング処理を説明するフローチャートである。

図25は、ウォーターマークの構成を説明する図である。

図26は、コンテンツのフォーマットの例を示す図である。

図27は、公開鍵証明書の例を示す図である。

25 図28は、コンテンツの配布を説明する図である。

図29は、クライアントのコンテンツのチェックアウト処理を説明するフローチャートである。

図30は、タグによる有効化キープロックをたどる例を説明する図である。

図31は、有効化キープロックの構成例を示す図である。

図32は、マークの構成を説明する図である。

図33は、クライアントのライセンス買い取り処理を説明するフローチャート  
5 である。

図34は、ライセンスサーバのライセンス買い取り処理を説明するフローチャートである。

図35は、マークの構成例を示す図である。

図36は、クライアントの証明書の登録処理を説明するフローチャートである。

10 図37は、コンテンツサーバの証明書登録処理を説明するフローチャートである。

図38は、グループの証明書の例を示す図である。

図39は、グルーピングが行われている場合におけるコンテンツサーバの処理を説明するフローチャートである。

15 図40は、コンテンツキーの暗号化の例を示す図である。

図41は、グループに属するクライアントの処理を説明するフローチャートである。

図42は、他のクライアントにライセンスをチェックアウトするクライアントの処理を説明するフローチャートである。

20 図43は、他のクライアントからライセンスのチェックアウトを受けるクライアントの処理を説明するフローチャートである。

図44は、ライセンスのチェックアウトを受けたクライアントの再生処理を説明するフローチャートである。

25 図45は、他のクライアントからライセンスのチェックインを受けるクライアントの処理を説明するフローチャートである。

図46は、他のクライアントにライセンスをチェックインするクライアントの処理を説明するフローチャートである。

図47は、MACの生成を説明する図である。

図48は、ICV生成キーの復号処理を説明するフローチャートである。

図49は、ICV生成キーの他の復号処理を説明する図である。

図50Aは、ICVによるライセンスのコピーの管理を説明する図である。

5 図50Bは、ICVによるライセンスのコピーの管理を説明する図である。

図51は、ライセンスの管理を説明する図である。

#### 発明を実施するための最良の形態

図1は、本発明を適用したコンテンツ提供システムの構成を示している。イン

10 ターネット2には、クライアント1-1, 1-2(以下、これらのクライアントを個々に区別する必要がない場合、単にクライアント1と称する)が接続されている。この例においては、クライアントが2台のみ示されているが、インターネット2には、任意の台数のクライアントが接続される。

また、インターネット2には、クライアント1に対してコンテンツを提供するコンテンツサーバ3、コンテンツサーバ3が提供するコンテンツを利用するのに必要なライセンスをクライアント1に対して付与するライセンスサーバ4、およびクライアント1がライセンスを受け取った場合に、そのクライアント1に対して課金処理を行う課金サーバ5が接続されている。

20 これらのコンテンツサーバ3、ライセンスサーバ4、および課金サーバ5も、任意の台数、インターネット2に接続される。

図2はクライアント1の構成を表している。

図2において、CPU(Central Processing Unit)21は、ROM(Read Only Memory)22に記憶されているプログラム、または記憶部28からRAM(Random Access Memory)23にロードされたプログラムに従って各種の処理を実行する。タイマ20は、計時動作を行い、時刻情報をCPU21に供給する。RAM23にはまた、CPU21が各種の処理を実行する上において必要なデータなども適宜記憶される。

暗号化復号部 24 は、コンテンツデータを暗号化するとともに、既に暗号化されているコンテンツデータを復号する処理を行う。コーデック部 25 は、例えば、ATRAC (Adaptive Transform Acoustic Coding) 3 方式などでコンテンツデータをエンコードし、入出力インターフェース 32 を介してドライブ 30 に接続されている半導体メモリ 44 に供給し、記録させる。あるいはまた、コーデック部 25 は、ドライブ 30 を介して半導体メモリ 44 より読み出した、エンコードされているデータをデコードする。

半導体メモリ 44 は、例えば、メモリスティック (商標) などにより構成される。

10 CPU 21、ROM 22、RAM 23、暗号化復号部 24、およびコーデック部 25 は、バス 31 を介して相互に接続されている。このバス 31 にはまた、入出力インターフェース 32 も接続されている。

入出力インターフェース 32 には、キーボード、マウスなどよりなる入力部 26、CRT、LCD などよりなるディスプレイ、並びにスピーカなどよりなる出力部 27、ハードディスクなどより構成される記憶部 28、モ뎀、ターミナルアダプタなどより構成される通信部 29 が接続されている。通信部 29 は、インターネット 2 を介しての通信処理を行う。通信部 29 はまた、他のクライアントとの間で、アナログ信号またはデジタル信号の通信処理を行う。

入出力インターフェース 32 にはまた、必要に応じてドライブ 30 が接続され、

20 磁気ディスク 41、光ディスク 42、光磁気ディスク 43、或いは半導体メモリ 44 などが適宜装着され、それらから読み出されたコンピュータプログラムが、必要に応じて記憶部 28 にインストールされる。

なお、図示は省略するが、コンテンツサーバ 3、ライセンスサーバ 4、課金サーバ 5 も、図 2 に示したクライアント 1 と基本的に同様の構成を有するコンピュータにより構成される。そこで、以下の説明においては、図 2 の構成は、コンテンツサーバ 3、ライセンスサーバ 4、課金サーバ 5 などの構成としても引用される。

次に、図3のフローチャートを参照して、クライアント1がコンテンツサーバ3からコンテンツの提供を受ける処理について説明する。

ユーザが、入力部26を操作することでコンテンツサーバ3に対するアクセスを指令すると、CPU21は、ステップS1において、通信部29を制御し、インターネット2を介してコンテンツサーバ3にアクセスさせる。ステップS2において、ユーザが、入力部26を操作して、提供を受けるコンテンツを指定すると、CPU21は、この指定情報を受け取り、通信部29から、インターネット2を介してコンテンツサーバ3に、指定されたコンテンツを通知する。図4のフローチャートを参照して後述するように、この通知を受けたコンテンツサーバ3は、暗号化されたコンテンツデータを送信してくるので、ステップS3において、CPU21は、通信部29を介して、このコンテンツデータを受信すると、ステップS4において、その暗号化されているコンテンツデータを記憶部28を構成するハードディスクに供給し、記憶させる。

次に、図4のフローチャートを参照して、クライアント1の以上の処理に対応するコンテンツサーバ3のコンテンツ提供処理について説明する。なお、以下の説明において、図2のクライアント1の構成は、コンテンツサーバ3の構成としても引用される。

ステップS21において、コンテンツサーバ3のCPU21は、インターネット2から通信部29を介してクライアント1よりアクセスを受けるまで待機し、アクセスを受けたと判定したとき、ステップS22に進み、クライアント1から送信されてきたコンテンツを指定する情報を取り込む。このコンテンツを指定する情報は、クライアント1が、図3のステップS2において通知してきた情報である。

ステップS23において、コンテンツサーバ3のCPU21は、記憶部28に記憶されているコンテンツデータの中から、ステップS22の処理で取り込まれた情報で指定されたコンテンツを読み出す。CPU21は、ステップS24にお

いて、記憶部 28 から読み出されたコンテンツデータを、暗号化復号部 24 に供給し、コンテンツキー  $K_c$  を用いて暗号化させる。

記憶部 28 に記憶されているコンテンツデータは、コーデック部 25 により、既に ATRAC 3 方式によりエンコードされているので、このエンコードされてい  
5 るコンテンツデータが暗号化されることになる。

なお、もちろん、記憶部 28 に予め暗号化した状態でコンテンツデータを記憶させることができる。この場合には、ステップ S 24 の処理は省略することが可能である。

次に、ステップ S 25 において、コンテンツサーバ 3 の CPU 21 は、暗号化  
10 したコンテンツデータを伝送するフォーマットを構成するヘッダに、暗号化されているコンテンツを復号するのに必要なキー情報（図 5 を参照して後述する EKB (Enabling Key Block) と KEKBC (Kc) ）と、コンテンツを利用するのに必要なライセンスを識別するためのライセンス ID を付加する。そして、ステップ S 26 において、コンテンツサーバ 3 の CPU 21 は、ステップ S 24 の処理で暗号化したコンテンツと、ステップ S 25 の処理でキーとライセンス ID を付加したヘッダとをフォーマット化したデータを、通信部 29 から、インターネット 2 を介して、アクセスしてきたクライアント 1 に送信する。

図 5 は、このようにして、コンテンツサーバ 3 からクライアント 1 にコンテンツが供給される場合のフォーマットの構成を表している。同図に示されるように、  
20 このフォーマットは、ヘッダ (Header) とデータ (Data) とにより構成される。

ヘッダには、コンテンツ情報 (Content information) 、URL (Uniform Resource Locator) 、ライセンス ID (License ID) 、イネーブリングキーブロック (有効化キーブロック) (EKB (Enabling Key Block)) および、  
25 EKB から生成されたキー  $K_{EKBC}$  を用いて暗号化されたコンテンツキー  $K_c$  としてのデータ  $K_{EKBC}$  ( $K_c$ ) が配置されている。なお、EKB については、図 15 A および図 15 B を参照して後述する。

コンテンツ情報には、データとしてフォーマット化されているコンテンツデータを識別するための識別情報としてのコンテンツ ID (CID) 、そのコンテンツのコーデックの方式などの情報が含まれている。

URL は、ライセンス ID で規定されるライセンスを取得するときアクセスするアドレス情報であり、図 1 のシステムの場合、具体的には、ライセンスを受けるために必要なライセンスサーバ 4 のアドレスである。ライセンス ID は、データとして記録されているコンテンツを利用するとき必要とされるライセンスを識別するものである。

データは、任意の数の暗号化ブロック (Encryption Block) により構成される。各暗号化ブロックは、イニシャルベクトル (IV (Initial Vector) ) 、シード (Seed) 、およびコンテンツデータをキー  $K'c$  で暗号化したデータ  $EK'c(data)$  により構成されている。

キー  $K'c$  は、次式により示されるように、コンテンツキー  $Kc$  と、乱数で設定される値  $Seed$  をハッシュ関数に適用して演算された値により構成される。

$$K'c = \text{Hash}(Kc, Seed)$$

イニシャルベクトル IV とシード Seed は、各暗号化ブロック毎に異なる値に設定される。

この暗号化は、コンテンツのデータを 8 バイト単位で区分して、8 バイト毎に行われる。後段の 8 バイトの暗号化は、前段の 8 バイトの暗号化の結果を利用して行われる CBC (Cipher Block Chaining) モードで行われる。

CBC モードの場合、最初の 8 バイトのコンテンツデータを暗号化するとき、その前段の 8 バイトの暗号化結果が存在しないため、最初の 8 バイトのコンテンツデータを暗号化するときは、イニシャルベクトル IV を初期値として暗号化が行われる。

この CBC モードによる暗号化を行うことで、1 つの暗号化ブロックが解読されたとしても、その影響が、他の暗号化ブロックにおよぶことが抑制される。

なお、この暗号化については、図 4.7 を参照して、後に詳述する。

また、暗号方式についてはこれに限らず、単にコンテンツキーKcでコンテンツデータを暗号化しても良い。

以上のようにして、クライアント1は、コンテンツサーバ3からコンテンツを無料で、自由に取得することができる。従って、コンテンツそのものは、大量に、

5 配布することが可能となる。

しかしながら、各クライアント1は、取得したコンテンツを利用するとき、ライセンスを保持している必要がある。そこで、図6を参照して、クライアント1がコンテンツを再生する場合の処理について説明する。

ステップS41において、クライアント1のCPU21は、ユーザが入力部210を操作することで指示したコンテンツの識別情報(CID)を取得する。この識別情報は、例えば、コンテンツのタイトルや、記憶されている各コンテンツ毎に付与されている番号などにより構成される。

そして、CPU21は、コンテンツが指示されると、そのコンテンツに対応するライセンスID(そのコンテンツを使用するのに必要なライセンスのID)を読み取る。このライセンスIDは、図5に示されるように、暗号化されているコンテンツデータのヘッダに記述されているものである。

次に、ステップS42に進み、CPU21は、ステップS41で読み取られたライセンスIDに対応するライセンスが、クライアント1により既に取得され、記憶部28に記憶されているか否かを判定する。まだ、ライセンスが取得されていない場合には、ステップS43に進み、CPU21は、ライセンス取得処理を実行する。このライセンス取得処理の詳細は、図7のフローチャートを参照して後述する。

ステップS42において、ライセンスが既に取得されていると判定された場合、または、ステップS43において、ライセンス取得処理が実行された結果、ライセンスが取得された場合、ステップS44に進み、CPU21は、取得されているライセンスは有効期限内のものであるか否かを判定する。ライセンスが有効期限内のものであるか否かは、ライセンスの内容として規定されている期限(後述

する図 8 参照) と、タイマ 20 により計時されている現在日時と比較することで判断される。ライセンスの有効期限が既に満了していると判定された場合、CPU 21 は、ステップ S 45 に進み、ライセンス更新処理を実行する。このライセンス更新処理の詳細は、図 10 のフローチャートを参照して後述する。

5 ステップ S 44において、ライセンスはまだ有効期限内であると判定された場合、または、ステップ S 45において、ライセンスが更新された場合、ステップ S 46 に進み、CPU 21 は、暗号化されているコンテンツデータを記憶部 28 から読み出し、RAM 23 に格納させる。そして、ステップ S 47において、CPU 21 は、RAM 23 に記憶された暗号化ブロックのデータを、図 5 のデータ 10 に配置されている暗号化ブロック単位で、暗号化復号部 24 に供給し、コンテンツキー  $K_c$  を用いて復号させる。

コンテンツキー  $K_c$  を得る方法の具体例は、図 15A および図 15B を参照して後述するが、デバイスノードキー (DNK(Device Node Key)) を用いて、EKB (図 5) に含まれるキー  $K_{EKBC}$  を得ることができ、そのキー  $K_{EKBC}$  を用いて、データ  $K_{EKBC}$  ( $K_c$ ) (図 5) から、コンテンツキー  $K_c$  を得ることができる。

CPU 21 は、さらに、ステップ S 48 において、暗号化復号部 24 により復号されたコンテンツデータをコーデック部 25 に供給し、デコードさせる。そして、コーデック部 25 によりデコードされたデータを、CPU 21 は、入出力インターフェース 32 から出力部 27 に供給し、D/A 変換させ、スピーカから出力 20 させる。

次に、図 7 のフローチャートを参照して、図 6 のステップ S 43 で行われるライセンス取得処理の詳細について説明する。

クライアント 1 は、事前にライセンスサーバにアクセスして登録処理を行うことにより、リーフ ID、DNK(Device Node Key)、クライアント 1 の秘密鍵・公開鍵のペア、ライセンスサーバの公開鍵、及び各公開鍵の証明書を含むサービスデータを取得しておく。クライアントの登録処理の詳細は図 23 を参照して後述する。

リーフ ID は、クライアント毎に割り当てられた識別情報を表し、DNK は、そのライセンスに対応する EKB (有効化キーブロック) に含まれる暗号化されているコンテンツキーKc を復号するのに必要なデバイスノードキーである (図 12 を参照して後述する)。

- 5 最初にステップ S 6 1において、CPU 2 1は、いま処理対象とされているライセンス ID に対応する URL を、図 5 に示すヘッダから取得する。上述したように、この URL は、やはりヘッダに記述されているライセンス ID に対応するライセンスを取得するときアクセスすべきアドレスである。そこで、ステップ S 6 2において、CPU 2 1は、ステップ S 6 1で取得した URL にアクセスする。
- 10 具体的には、通信部 2 9 によりインターネット 2 を介してライセンスサーバ 4 にアクセスが行われる。このとき、ライセンスサーバ 4 は、クライアント 1 に対して、購入するライセンス (コンテンツを使用するのに必要なライセンス) を指定するライセンス指定情報、並びにユーザ ID とパスワードの入力を要求してくる (後述する図 9 のステップ S 10 2)。CPU 2 1は、この要求を出力部 2 7 の表示部に表示させる。ユーザは、この表示に基づいて、入力部 2 6 を操作して、ライセンス指定情報、ユーザ ID、およびパスワードを入力する。なお、このユーザ ID とパスワードは、クライアント 1 のユーザが、インターネット 2 を介してライセンスサーバ 4 にアクセスし、事前に取得しておいたものである。
- 15 CPU 2 1は、ステップ S 6 3, S 6 4において、入力部 2 6 から入力されたライセンス指定情報を取り込むとともに、ユーザ ID とパスワードを取り込む。
- 20 CPU 2 1は、ステップ S 6 5において、通信部 2 9 を制御し、入力されたユーザ ID とパスワード、ライセンス指定情報、並びにサービスデータ (後述する) に含まれるリーフ ID を含むライセンス要求を、インターネット 2 を介してライセンスサーバ 4 に送信させる。
- 25 ライセンスサーバ 4 は、図 9 を参照して後述するように、ユーザ ID とパスワード、並びにライセンス指定情報に基づいてライセンスを送信してくる (ステッ

プ S 1 0 9 ) か、または、条件が満たされない場合には、ライセンスを送信してこない (ステップ S 1 1 2 ) 。

ステップ S 6 6 において、CPU 2 1 は、ライセンスサーバ 4 からライセンスが送信されてきたか否かを判定し、ライセンスが送信されてきた場合には、ステップ S 6 7 に進み、そのライセンスを記憶部 2 8 に供給し、記憶させる。

ステップ S 6 6 において、ライセンスが送信されて来ないと判定した場合、CPU 2 1 は、ステップ S 6 8 に進み、エラー処理を実行する。具体的には、CPU 2 1 は、コンテンツを利用するためのライセンスが得られないので、コンテンツの再生処理を禁止する。

10 以上のようにして、各クライアント 1 は、コンテンツデータに付随しているライセンス ID に対応するライセンスを取得して、初めて、そのコンテンツを使用することが可能となる。

なお、図 7 のライセンス取得処理は、各ユーザがコンテンツを取得する前に、予め行っておくようによくすることも可能である。

15 クライアント 1 に提供されるライセンスは、例えば、図 8 に示されるように、使用条件、リーフ ID 等を含んでいる。

使用条件には、そのライセンスに基づいて、コンテンツを使用することが可能な使用期限、そのライセンスに基づいて、コンテンツをダウンロードすることができるダウンロード期限、そのライセンスに基づいて、コンテンツをコピーすることが可能な回数 (許されるコピー回数) 、チェックアウト回数、最大チェックアウト回数、そのライセンスに基づいて、コンテンツを CD-R に記録することができる権利、PD (Portable Device) にコピーすることができる回数、ライセンスを所有権 (買い取り状態) に移行できる権利、使用ログをとる義務等を示す情報が含まれる。

25 次に、図 9 のフローチャートを参照して、図 7 のクライアント 1 のライセンス取得処理に対応して実行されるライセンスサーバ 4 のライセンス提供処理について

て説明する。なお、この場合においても、図2のクライアント1の構成は、ライセンスサーバ4の構成として引用される。

ステップS101において、ライセンスサーバ4のCPU21は、クライアント1よりアクセスを受けるまで待機し、アクセスを受けたとき、ステップS105に進み、アクセスしてきたクライアント1に対して、ユーザIDとパスワード、並びに、ライセンス指定情報の送信を要求する。上述したようにして、クライアント1から、図7のステップS65の処理で、ユーザIDとパスワード、リーフID並びにライセンス指定情報（ライセンスID）が送信されてきたとき、ライセンスサーバ4のCPU21は、通信部29を介してこれを受信し、取り込む処理10を実行する。

そして、ライセンスサーバ4のCPU21は、ステップS103において、通信部29から課金サーバ5にアクセスし、ユーザIDとパスワードに対応するユーザの与信処理を要求する。課金サーバ5は、インターネット2を介してライセンスサーバ4から与信処理の要求を受けると、そのユーザIDとパスワードに対応するユーザの過去の支払い履歴などを調査し、そのユーザが、過去にライセンスの対価の不払いの実績があるか否かなどを調べ、そのような実績がない場合には、ライセンスの付与を許容する与信結果を送信し、不払いの実績などがある場合には、ライセンス付与の不許可の与信結果を送信する。

ステップS104において、ライセンスサーバ4のCPU21は、課金サーバ5からの与信結果が、ライセンスを付与することを許容する与信結果であるか否かを判定し、ライセンスの付与が許容されている場合には、ステップS105に進み、ステップS102の処理で取り込まれたライセンス指定情報に対応するライセンスを、記憶部28に記憶されているライセンスの中から取り出す。記憶部28に記憶されているライセンスは、あらかじめライセンスID、バージョン、25作成日時、有効期限等の情報が記述されている。ステップS106において、CPU21は、そのライセンスに受信したリーフIDを付加する。さらに、ステップS107において、CPU21は、ステップS105で選択されたライセンス

に対応づけられている使用条件を選択する。あるいはまた、ステップS102の処理で、ユーザから使用条件が指定された場合には、その使用条件が必要に応じて、予め用意されている使用条件に付加される。CPU21は、選択された使用条件をライセンスに付加する。

5   ステップS108において、CPU21はライセンスサーバの秘密鍵によりライセンスに署名し、これにより、図8に示されるような構成のライセンスが生成される。

次に、ステップS109に進み、ライセンスサーバ4のCPU21は、そのライセンス（図8に示される構成を有する）を、通信部29からインターネット210を介してクライアント1に送信させる。

ステップS110においてライセンスサーバ4のCPU21は、ステップS109の処理で、いま送信したライセンス（使用条件、リーフIDを含む）を、ステップS102の処理で取り込まれたユーザIDとパスワードに対応して、記憶部28に記憶させる。さらに、ステップS111において、CPU21は、課金処理を実行する。具体的には、CPU21は、通信部29から課金サーバ5に、そのユーザIDとパスワードに対応するユーザに対する課金処理を要求する。課金サーバ5は、この課金の要求に基づいて、そのユーザに対する課金処理を実行する。上述したように、この課金処理に対して、そのユーザが支払いを行わなかったような場合には、以後、そのユーザは、ライセンスの付与を要求したとしても、ライセンスを受けることができないことになる。

すなわち、この場合には、課金サーバ5からライセンスの付与を不許可とする与信結果が送信されてくるので、ステップS104からステップS112に進み、CPU21は、エラー処理を実行する。具体的には、ライセンスサーバ4のCPU21は、通信部29を制御してアクセスしてきたクライアント1に対して、ライセンスを付与することができない旨のメッセージを出力し、処理を終了させる。

この場合、上述したように、そのクライアント1はライセンスを受けることができないので、そのコンテンツを利用すること（暗号を復号すること）ができないことになる。

図10は、図6のステップS45におけるライセンス更新処理の詳細を表している。図10のステップS131乃至ステップS135の処理は、図7のステップS61乃至ステップS65の処理と基本的に同様の処理である。ただし、ステップS133において、CPU21は、購入するライセンスではなく、更新するライセンスのライセンスIDを取り込む。そして、ステップS135において、CPU21は、ユーザIDとパスワードとともに、更新するライセンスのライセンスIDを、ライセンスサーバ4に送信する。

ステップS135の送信処理に対応して、ライセンスサーバ4は、後述するように、使用条件を提示してくる（図11のステップS153）。そこで、クライアント1のCPU21は、ステップS136において、ライセンスサーバ4からの使用条件の提示を受信し、これを出力部27に出力し、表示させる。ユーザは、15 入力部26を操作して、この使用条件の中から所定の使用条件を選択したり、所定の使用条件を新たに追加したりする。ステップS137でCPU21は、以上のようにして選択された使用条件（ライセンスを更新する条件）を購入するための申し込みをライセンスサーバ4に送信する。この申し込みに対応して、後述するようにライセンスサーバ4は、最終的な使用条件を送信してくる（図11のステップS154）。そこで、ステップS138において、クライアント1のCPU21は、ライセンスサーバ4からの使用条件を取得し、ステップS139において、その使用条件を記憶部28にすでに記憶されている対応するライセンスの使用条件として更新する。

図11は、以上のクライアント1のライセンス更新処理に対応して、ライセンスサーバ4が実行するライセンス更新処理を表している。

最初に、ステップS151において、ライセンスサーバ4のCPU21は、クライアント1からのアクセスを受けると、ステップS152において、クライア

ント 1 がステップ S 1 3 5 で送信したライセンス指定情報をライセンス更新要求情報とともに受信する。

ステップ S 1 5 3 において、CPU 2 1 は、ライセンスの更新要求を受信すると、そのライセンスに対応する使用条件（更新する使用条件）を、記憶部 2 8 から読み出し、クライアント 1 に送信する。

この提示に対して、上述したように、クライアント 1 から使用条件の購入が図 1 0 のステップ S 1 3 7 の処理で申し込みされると、ステップ S 1 5 4 において、ライセンスサーバ 4 の CPU 2 1 は、申し込みされた使用条件に対応するデータを生成し、ステップ S 1 5 4 において、クライアント 1 に送信する。クライアント 1 は、上述したように、ステップ S 1 3 9 の処理で受信した使用条件を用いて、すでに登録されているライセンスの使用条件を更新する。

本発明においては、図 1 2 に示されるように、ブロードキャストインクリプション（Broadcast Encryption）方式の原理に基づいて、デバイスとライセンスのキーが管理される（特開 2001-352321 号公報参照）。キーは、階層ツリー構造とされ、最下段のリーフ（leaf）が個々のデバイスのキーに対応する。図 1 2 の例の場合、番号 0 から番号 1 5 までの 1 6 個のデバイスまたはライセンスに対応するキーが生成される。

各キーは、図中丸印で示されるツリー構造の各ノードに対応して規定される。この例では、最上段のルートノードに対応してルートキー KR が、2 段目のノードに対応してキー K 0 , K 1 が、3 段目のノードに対応してキー K 0 0 乃至 K 1 が、第 4 段目のノードに対応してキー K 0 0 0 乃至キー K 1 1 1 が、それぞれ対応されている。そして、最下段のノードとしてのリーフ（デバイスノード）に、キー K 0 0 0 0 乃至 K 1 1 1 が、それぞれ対応している。

階層構造とされているため、例えば、キー K 0 0 1 0 とキー 0 0 1 1 の上位のキーは、K 0 0 1 とされ、キー K 0 0 0 とキー K 0 0 1 の上位のキーは、K 0 0 とされている。以下同様に、キー K 0 0 とキー K 0 1 の上位のキーは、K 0 とされ、キー K 0 とキー K 1 の上位のキーは、KR とされている。

コンテンツを利用するキーは、最下段のリーフから、最上段のルートノードまでの 1 つのパスの各ノードに対応するキーで管理される。例えば、番号 3 のノード（リーフ ID）に対応するライセンスに基づき、コンテンツを利用するキーは、キー K0011, K001, K00, K0, KR を含むパスの各キーで管理される。

5 る。

本発明のシステムにおいては、図 13 に示されるように、図 12 の原理に基づいて構成されるキーシステムで、デバイスのキーとライセンスのキーの管理が行われる。図 13 の例では、8 + 24 + 32 段のノードがツリー構造とされ、ルートノードから下位の 8 段までの各ノードにカテゴリが対応される。ここにおける 10 カテゴリとは、例えばメモリスティックなどの半導体メモリを使用する機器のカテゴリ、デジタル放送を受信する機器のカテゴリといったカテゴリを意味する。そして、このカテゴリノードのうちの 1 つのノードに、ライセンスを管理するシステムとして本システム（T システムと称する）が対応する。

すなわち、この T システムのノードよりさらに下の階層の 24 段のノードに対応するキーにより、ライセンスが対応される。この例の場合、これにより、2 の 24 乗（約 16 メガ）のライセンスを規定することができる。さらに、最も下側の 32 段の階層により、2 の 32 乗（約 4 ギガ）のユーザ（あるいはクライアント 1）を規定することができる。最下段の 32 段のノードに対応するリーフからルートノードまでのパスの各ノードに対応するキーが、DNK（Device Node 20 Key）を構成し、最下段のリーフに対応する ID がリーフ ID とされる。

各デバイスやライセンスのキーは、64 (= 8 + 24 + 32) 段の各ノードで構成されるパスの内の 1 つに対応される。例えば、コンテンツを暗号化したコンテンツキーは、対応するライセンスに割り当てられたパスを構成するノードに対応するキーを用いて暗号化される。上位の階層のキーは、その直近の下位の階層 25 のキーを用いて暗号化され、EKB（図 15A および図 15B を参照して後述する）内に配置される。DNK は、EKB 内には配置されず、サービスデータに記述され、ユーザのクライアント 1 に与えられる。クライアント 1 は、サービスデ

ータに記述されているDNKを用いて、コンテンツデータとともに配布されるEK  
B（図15Aおよび図15B）内に記述されている直近の上位の階層のキーを復号し、復号して得たキーを用いて、EK B内に記述されているさらにその上の階層のキーを復号する。以上の処理を順次行うことで、クライアント1は、その  
5 パスに属するすべてのキーを得ることができる。

図14に階層ツリー構造のカテゴリの分類の具体的な例を示す。図14において、階層ツリー構造の最上段には、ルートキーKR2301が設定され、以下の中間段にはノードキー2302が設定され、最下段には、リーフキー2303が設定される。各デバイスは個々のリーフキーと、リーフキーからルートキーに至  
10 る一連のノードキー、ルートキーを保有する。

最上段から第M段目（図13の例では、M=8）の所定のノードがカテゴリノ  
ード2304として設定される。すなわち第M段目のノードの各々が特定カテゴリのデバイス設定ノードとされる。第M段の1つのノードを頂点としてM+1段  
以下ノード、リーフは、そのカテゴリに含まれるデバイスに関するノードおよ  
15 びリーフとされる。

例えば図14の第M段目の1つのノード2305にはカテゴリ[メモリスティック（商標）]が設定され、このノード以下に連なるノード、リーフはメモリスティックを使用した様々なデバイスを含むカテゴリ専用のノードまたはリーフとして設定される。すなわち、ノード2305以下が、メモリスティックのカテゴリに定義されるデバイスの関連ノード、およびリーフの集合として定義される。  
20

さらに、M段から数段分下位の段をサブカテゴリノード2306として設定することができる。図14の例では、カテゴリ[メモリスティック]ノード2305の2段下のノードに、メモリスティックを使用したデバイスのカテゴリに含まれるサブカテゴリノードとして、[再生専用器]のノード2306が設定されて  
25 いる。さらに、サブカテゴリノードである再生専用器のノード2306以下に、再生専用器のカテゴリに含まれる音楽再生機能付き電話のノード2307が設定

され、さらにその下位に、音楽再生機能付き電話のカテゴリに含まれる [P H S] ノード 2308 と、[携帯電話] ノード 2309 が設定されている。

さらに、カテゴリ、サブカテゴリは、デバイスの種類のみならず、例えばある  
5 メーカー、コンテンツプロバイダ、決済機関等が独自に管理するノード、すなわ  
ち処理単位、管轄単位、あるいは提供サービス単位等、任意の単位（これらを総  
称して以下、エンティティと呼ぶ）で設定することが可能である。例えば 1 つの  
10 カテゴリノードをゲーム機器メーカーの販売するゲーム機器 XYZ 専用の頂点ノ  
ードとして設定すれば、メーカーの販売するゲーム機器 XYZ に、その頂点ノー  
ド以下の下段のノードキー、リーフキーを格納して販売することが可能となり、  
15 その後、暗号化コンテンツの配信、あるいは各種キーの配信、更新処理を、その  
頂点ノードキー以下のノードキー、リーフキーによって構成される有効化キープ  
ロック (EKB) を生成して配信し、頂点ノード以下のデバイスに対してのみ利  
用可能なデータが配信可能となる。

このように、1 つのノードを頂点として、以下のノードをその頂点ノードに定  
15 義されたカテゴリ、あるいはサブカテゴリの関連ノードとして設定する構成とす  
ることにより、カテゴリ段、あるいはサブカテゴリ段の 1 つの頂点ノードを管理  
するメーカー、コンテンツプロバイダ等がそのノードを頂点とする有効化キープ  
ロック (EKB) を独自に生成して、頂点ノード以下に属するデバイスに配信す  
る構成が可能となり、頂点ノードに属しない他のカテゴリのノードに属するデバ  
20 イスには全く影響を及ぼさずにキー更新を実行することができる。

例えば、図 12 に示されるツリー構造において、1 つのグループに含まれる 4  
つのデバイス 0, 1, 2, 3 はノードキーとして共通のキー K00, K0, KR  
を保有する。このノードキー共有構成を利用することにより、共通のコンテンツ  
キーをデバイス 0, 1, 2, 3 のみに提供することが可能となる。たとえば、共  
25 通に保有するノードキー K00 自体をコンテンツキーとして設定すれば、新たな  
鍵送付を実行することなくデバイス 0, 1, 2, 3 のみが共通のコンテンツキー  
の設定が可能である。また、新たなコンテンツキー Kcon をノードキー K00

で暗号化した値  $E_{nc}(K_{00}, K_{con})$  を、ネットワークを介してあるいは記録媒体に格納してデバイス 0, 1, 2, 3 に配布すれば、デバイス 0, 1, 2, 3 のみが、それぞれのデバイスにおいて保有する共有ノードキー  $K_{00}$  を用いて暗号  $E_{nc}(K_{00}, K_{con})$  を解いてコンテンツキー  $K_{con}$  を得ることが可能となる。なお、 $E_{nc}(K_a, K_b)$  は  $K_b$  を  $K_a$  によって暗号化したデータであることを示す。

また、ある時点  $t$  において、デバイス 3 の所有する鍵  $K_{0011}, K_{001}, K_{00}, K_{0}, KR$  が攻撃者（ハッカー）により解析されて露呈したことが発覚した場合、それ以降、システム（デバイス 0, 1, 2, 3 のグループ）で送受信されるデータを守るために、デバイス 3 をシステムから切り離す必要がある。そのためには、ノードキー  $K_{001}, K_{00}, K_{0}, KR$  を、それぞれ新たな鍵  $K(t)_{001}, K(t)_{00}, K(t)_{0}, K(t)_{R}$  に更新し、デバイス 0, 1, 2 にその更新キーを伝える必要がある。ここで、 $K(t)_{aaa}$  は、鍵  $K_{aaa}$  の世代（Generation） $t$  の更新キーであることを示す。

更新キーの配布処理について説明する。キーの更新は、例えば、図 15A に示す有効化キーブロック（EKB : Enabling Key Block）と呼ばれるブロックデータによって構成されるテーブルを、ネットワークを介して、あるいは記録媒体に格納してデバイス 0, 1, 2 に供給することによって実行される。なお、有効化キーブロック（EKB）は、図 12 に示されるようなツリー構造を構成する各リーフ（最下段のノード）に対応するデバイスに、新たに更新されたキーを配布するための暗号化キーによって構成される。有効化キーブロック（EKB）は、キー更新ブロック（KRB : Key Renewal Block）と呼ばれることもある。

図 15A に示す有効化キーブロック（EKB）は、ノードキーの更新の必要なデバイスのみが更新可能なデータ構成を持つブロックデータとして構成される。図 15A の例は、図 12 に示すツリー構造中のデバイス 0, 1, 2 において、世代  $t$  の更新ノードキーを配布することを目的として形成されたブロックデータである。図 12 から明らかのように、デバイス 0, デバイス 1 は、更新ノードキー

として  $K(t)_{00}$ 、 $K(t)_0$ 、 $K(t)_R$  が必要であり、デバイス 2 は、更新ノードキーとして  $K(t)_{001}$ 、 $K(t)_{00}$ 、 $K(t)_0$ 、 $K(t)_R$  が必要である。

図 15 A の EKB に示されるように、EKB には複数の暗号化キーが含まれる。

5 図 15 A の最下段の暗号化キーは、 $Enc(K0010, K(t)001)$  である。これはデバイス 2 の持つリーフキー  $K0010$  によって暗号化された更新ノードキー  $K(t)001$  であり、デバイス 2 は、自身の持つリーフキー  $K0010$  によってこの暗号化キーを復号し、更新ノードキー  $K(t)001$  を得ることができる。また、復号により得た更新ノードキー  $K(t)001$  を用いて、図 1  
10 15 A の下から 2 段目の暗号化キー  $Enc(K(t)001, K(t)00)$  が復号可能となり、更新ノードキー  $K(t)00$  を得ることができる。

以下順次、図 15 A の上から 2 段目の暗号化キー  $Enc(K(t)00, K(t)0)$  を復号することで、更新ノードキー  $K(t)0$  が得られ、これを用いて、図 15 A の上から 1 段目の暗号化キー  $Enc(K(t)0, K(t)_R)$  を復号することで、更新ルートキー  $K(t)_R$  が得られる。

一方、ノードキー  $K000$  は更新する対象に含まれておらず、ノード 0, 1 が、更新ノードキーとして必要なのは、 $K(t)00$ 、 $K(t)0$ 、 $K(t)_R$  である。ノード 0, 1 は、デバイスノードキーに含まれるノードキー  $K000$  を用いて、図 15 A の上から 3 段目の暗号化キー  $Enc(K000, K(t)00)$  を復号することで更新ノードキー  $K(t)00$  を取得し、以下順次、図 15 A の上から 2 段目の暗号化キー  $Enc(K(t)00, K(t)0)$  を復号することで、更新ノードキー  $K(t)0$  を得、図 15 A の上から 1 段目の暗号化キー  $Enc(K(t)0, K(t)_R)$  を復号することで、更新ルートキー  $K(t)_R$  を得る。このようにして、デバイス 0, 1, 2 は更新したキー  $K(t)_R$  を得ることができる。

なお、図 15 A のインデックスは、図の右側の暗号化キーを復号するための復号キーとして使用するノードキー、リーフキーの絶対番地を示す。

図12に示すツリー構造の上位段のノードキーK(t)0,K(t)Rの更新が不要であり、ノードキーK00のみの更新処理が必要である場合には、図15Bの有効化キーブロック(EKB)を用いることで、更新ノードキーK(t)00をデバイス0,1,2に配布することができる。

5 図15Bに示すEKBは、例えば特定のグループにおいて共有する新たなコンテンツキーを配布する場合に利用可能である。具体例として、図12に点線で示すグループ内のデバイス0,1,2,3がある記録媒体を用いており、新たな共通のコンテンツキーK(t)conが必要であるとする。このとき、デバイス0,1,2,3の共通のノードキーK00を更新したK(t)00を用いて新たな共  
10 通の更新コンテンツキーK(t)conを暗号化したデータEnc(K(t)00,K(t)con)が、図15Bに示されるEKBとともに配布される。この配布により、デバイス4など、その他のグループの機器が復号することができないデータとしての配布が可能となる。

すなわち、デバイス0,1,2はEKBを処理して得たキーK(t)00を用  
15 いて暗号文を復号すれば、t時点でのコンテンツキーK(t)conを得ることが可能になる。

図16に、t時点でのコンテンツキーK(t)conを得る処理例として、K(t)00を用いて新たな共通のコンテンツキーK(t)conを暗号化したデータEnc(K(t)00,K(t)con)と、図15Bに示すEKBとを記  
20 録媒体を介して受領したデバイス0の処理を示す。すなわちこの例は、EKBによる暗号化メッセージデータをコンテンツキーK(t)conとした例である。

図16に示すように、デバイス0は、記録媒体に格納されている世代t時点のEKBと、自分があらかじめ格納しているDNKに含まれるノードキーK000を用いて、上述した場合と同様のEKB処理により、ノードキーK(t)00を生成する。さらに、デバイス0は、復号した更新ノードキーK(t)00を用いて、更新コンテンツキーK(t)conを復号して、後にそれを使用するために自分だけが持つリーフキーK0000で暗号化して格納する。

図17に有効化キーブロック（EKB）のフォーマット例を示す。バージョン601は、有効化キーブロック（EKB）のバージョンを示す識別子である。なお、バージョンは、最新のEKBを識別する機能と、コンテンツとの対応関係を示す機能を持つ。デプスは、有効化キーブロック（EKB）の配布先のデバイス5に対する階層ツリーの階層数を示す。データポインタ603は、有効化キーブロック（EKB）中のデータ部606の位置を示すポインタであり、タグポインタ604はタグ部607の位置、署名ポインタ605は署名608の位置を示すポインタである。

データ部606は、例えば更新するノードキーを暗号化したデータを格納する。10 例えば図16に示すような更新されたノードキーに関する各暗号化キー等を格納する。

タグ部607は、データ部606に格納された暗号化されたノードキー、リーフキーの位置関係を示すタグである。このタグの付与ルールを、図18を用いて説明する。

15 図18では、データとして先に図15Aで説明した有効化キーブロック（EKB）を送付する例を示している。この時のデータは、図18のテーブルに示すようになる。このときの暗号化キーに含まれるトップノードのアドレスをトップノードアドレスとする。この例の場合は、ルートキーの更新キーK(t)Rが含まれているので、トップノードアドレスはKRとなる。このとき、例えば最上段20 のデータE\_nc(K(t)0, K(t)R)は、図18に示す階層ツリーに示す位置P0に対応する。次の段のデータは、E\_nc(K(t)00, K(t)0)であり、ツリー上では前のデータの左下の位置P00に対応する。ツリー構造の所定の位置から見て、その下に、データがある場合は、タグが0、ない場合はタグが1に設定される。タグは{左(L)タグ、右(R)タグ}として設定される。25 図18のテーブルの最上段のデータE\_nc(K(t)0, K(t)R)に対応する位置POの左下の位置PO0にはデータがあるので、Lタグ=0、右にはデー

タグがないので、Rタグ=1となる。以下、すべてのデータにタグが設定され、図18に示すデータ列、およびタグ列が構成される。

タグは、対応するデータE<sub>n</sub>c (K<sub>xxx</sub>, K<sub>yyy</sub>) が、ツリー構造のどこに位置しているのかを示すために設定されるものである。データ部606に格納されるキーデータE<sub>n</sub>c (K<sub>xxx</sub>, K<sub>yyy</sub>) . . . は、単純に暗号化されたキーの羅列データに過ぎないが、上述したタグによってデータとして格納された暗号化キーのツリー上の位置が判別可能となる。上述したタグを用いずに、先の図15Aおよび図15Bで説明した構成のように、暗号化データに対応させたノード・インデックスを用いて、例えば、

10 0 : E<sub>n</sub>c (K (t) 0, K (t) R)

00 : E<sub>n</sub>c (K (t) 00, K (t) 0)

000 : E<sub>n</sub>c (K ((t) 000, K (t) 00)

. . . のようなデータ構成とすることも可能であるが、このようなインデックスを用いた構成とすると、冗長なデータとなりデータ量が増大し、ネットワークを介する配信等においては好ましくない。これに対し、上述したタグをキー位置を示す索引データとして用いることにより、少ないデータ量でキー位置の判別が可能となる。

図17に戻って、EKBフォーマットについてさらに説明する。署名 (Signature) 608は、有効化キーブロック (EKB) を発行した例えば鍵管理センタ (ライセンスサーバ4) 、コンテンツロバイダ (コンテンツサーバ3) 、決済機関 (課金サーバ5) 等が実行する電子署名である。EKBを受領したデバイスは、署名検証によって正当な有効化キーブロック (EKB) 発行者が発行した有効化キーブロック (EKB) であることを確認する。

以上のようにして、ライセンスサーバ4から供給されたライセンスに基づいて、25 コンテンツサーバ3から供給されたコンテンツを利用する処理をまとめると、図19に示されるようになる。

すなわち、コンテンツサーバ3からクライアント1に対してコンテンツが提供されるとともに、ライセンスサーバ4からクライアント1にライセンスが供給される。コンテンツは、コンテンツキーKcにより、暗号化されており (Enc (Kc, Content) ) 、コンテンツキーKcは、ルートキーKR(EKB から得られるキーで  
5 あって、図5におけるキーK<sub>EKBC</sub>に対応する)で暗号化され (Enc (KR, Kc) ) 、EKBとともに、暗号化されたコンテンツに付加されてクライアント1に提供される。

図19の例におけるEKBには、例えば、図20に示されるように、DNKで復号可能なルートキーKRが含まれている (Enc (DNK, KR) ) 。従って、ク  
10 ライアント1は、サービスデータに含まれるDNKを利用して、EKBからルートキーKRを得ることができる。さらに、ルートキーKRを用いて、Enc (KR, Kc) からコンテンツキーKcを復号することができ、コンテンツキーKcを用いて、Enc (Kc, Content) からコンテンツを復号することができる。

このように、クライアント1にDNKを個別に割り当てるにより、図12、  
15 並びに図15Aおよび図15Bを参照して説明した原理に従って、個々のクライアント1のリボーカ (revoke) が可能になる。

また、ライセンスにリーフIDを付加して配布することにより、クライアント1において、サービスデータとライセンスの対応付けが行われることになり、ライセンスの不正コピーを防止することが可能になる。

20 また、クライアント用の証明書と秘密鍵をサービスデータとして配信するよう  
にすることで、エンドユーザも、これらを用いて不正コピーを防止可能なコンテンツを作成することが可能になる。

証明書と秘密鍵の利用については、図29のフローチャートを参照して後述する。

25 本発明においては、図13を参照して説明したように、カテゴリノードにライセンスを管理する本発明のコンテンツ配信システムと、各種のコンテンツを利用するデバイスのカテゴリが対応づけられるので、複数のDNKを同一のデバイス

に持たせることができる。その結果、異なるカテゴリのコンテンツを1つのデバイスで管理することが可能となる。

図21は、この関係を表している。すなわち、デバイスD1には、コンテンツ配信システムに基づいて、DNK1が割り当てられている、コンテンツ1を利用するライセンス及びサービスデータが記録される。同様に、このデバイスD1には、例えば、DNK2が割り当てられた、メモリスティックにCDからリッピングしたコンテンツ2を記録することができる。この場合、デバイスD1は、コンテンツ1とコンテンツ2という、異なるシステム（コンテンツ配信システムとデバイス管理システム）により配信されたコンテンツを同時に扱うことが可能となる。新たなDNKを割り当てるとき、既に割り当てられているDNKを削除するなどして、デバイスに1個のDNKだけを対応させるようにした場合、このようないことはできない。

また、図13における、例えば、下側の32階層の各三角形の1つ1つに、図22に示されるライセンスカテゴリ1とライセンスカテゴリ2を割り当てることにより、同一のカテゴリ内を、サブカテゴリを利用して、コンテンツのジャンル、レベル、販売店、配信サービス、コンテンツの出所、提供方法等の小さな集まりに分類して、管理することが可能となる。

図22の例においては、例えば、ライセンスカテゴリ1は、ジャズのジャンルに属し、ライセンスカテゴリ2は、ロックのジャンルに属する。ライセンスカテゴリ1には、ライセンスIDが1であるコンテンツ1とコンテンツ2を対応させ、それぞれユーザ1乃至ユーザ3に配布されている。ライセンスカテゴリ2は、ライセンスID2のコンテンツ3、コンテンツ4、およびコンテンツ5が含まれ、それぞれユーザ1とユーザ3に提供されている。

このように、本発明においては、カテゴリ毎に独立したキー管理が可能になる。また、DNKを、機器やメディアに予め埋め込むのではなく、ライセンスサーバ4により、登録処理を行う際に、各機器やメディアにダウンロードするようにすることで、ユーザによるキーの取得が可能なシステムを実現することができる。

この場合のクライアント 1 の登録処理について、図 2 3 を参照して説明する。

ステップ S 1 6 1において、クライアント 1 の C P U 2 1 は通信部 2 9 を制御してライセンスサーバ 4 にサービスデータ要求を送信する。ライセンスサーバ 4 の C P U 2 1 は、ステップ S 1 6 5 において、通信部 2 9 を介して入力されるサービスデータ要求を受信すると、S 1 6 6 において、通信部 2 9 を介してユーザ情報要求をクライアント 1 に送信する。

クライアント 1 の C P U 2 1 は、ステップ S 1 6 2 において、通信部 2 9 を介してユーザ情報要求を受信すると、出力部 2 7 を制御しディスプレイなどにユーザ情報の入力を促すメッセージを表示させる。ユーザがキーボードなどを操作することにより、入力部 2 6 からユーザ本人の個人情報や決済情報等のユーザ情報を入力すると、S 1 6 3 においてクライアント 1 の C P U 2 1 は、入力されたユーザ情報を、通信部 2 9 を介してライセンスサーバ 4 に送信する。

ライセンスサーバ 4 の C P U 2 1 は、ステップ S 1 6 7 において、通信部 2 9 を介してユーザ情報を受信すると、ステップ S 1 6 8 において、そのライセンスサーバ 4 に割り当てられたカテゴリのノード以下のリーフのうち、まだ割り当てられていないリーフをクライアント 1 に割り当て、そのリーフからライセンスサーバ 4 に割り当てられたカテゴリのノードまでのパス上のノードに割り当てられたノードキーの組をデバイスノードキーとして生成し、生成されたデバイスノードキー、クライアント 1 に割り当てられたリーフのリーフ ID、クライアント 1 の秘密鍵、クライアント 1 の秘密鍵・公開鍵のペア、ライセンスサーバの公開鍵、及び各公開鍵の証明書をまとめてサービスデータとして生成し、S 1 6 9 において通信部 2 9 を介してクライアントに生成されたサービスデータを送信すると共に、ドライブ 3 0 を制御してユーザ情報をリーフ ID と対応付けてハードディスク等の記録メディアに記録させる。

クライアント 1 の C P U 2 1 は、ステップ S 1 6 4 において、通信部 2 9 を介してサービスデータを受信すると、暗号化復号部 2 4 を制御して受信したサービ

データを暗号化し、ドライブ 30 を制御してハードディスク等の記録メディアに記録させる。

以上のようにして、ライセンスサーバ 4 はクライアント 1 及びそのユーザを登録し、クライアント 1 は所望のコンテンツ配信サービスを利用するためには必要な、

5 デバイスノードキーを含むサービスデータを受け取ることができる。

コンテンツは、それが作成された後、どのような使われ方をされようとも、その使われ方に関わりなく、全ての用途において、使用可能であるのが望ましい。

例えば、異なるコンテンツ配信サービス、あるいはドメインの使用状況が異なる場合においても、同一のコンテンツが使えることが望ましい。本発明においては、

10 このため、上述したように、各ユーザ（クライアント 1）に、認証局としてのライセンスサーバ 4 から秘密鍵と、それに対応する公開鍵の証明書

（certificates）が配布される。各ユーザは、その秘密鍵を用いて、署名（signature）を作成し、コンテンツに付加して、コンテンツの真正さ（integrity）を保証し、かつコンテンツの改竄防止を図ることができる。

15 この場合の処理の例について、図 24 のフローチャートを参照して説明する。

図 24 の処理は、ユーザが CD から再生したデータを記憶部 28 に記憶させるリッピング処理を説明するものである。

最初に、ステップ S171において、クライアント 1 の CPU21 は、通信部 29 を介して入力される CD の再生データを記録データとして取り込む。ステ

20 ップ S172において、CPU21 は、ステップ S171 の処理で取り込まれた記録データにウォーターマークが含まれているか否かを判定する。このウォーターマークは、3 ビットのコピー管理情報（CCI）と、1 ビットのトリガ

（Trigger）とにより構成され、コンテンツのデータの中に埋め込まれている。

CPU21 は、ウォーターマークが検出された場合には、ステップ S173 に進み、そのウォーターマークを抽出する処理を実行する。ウォーターマークが存在しない場合には、ステップ S173 の処理はスキップされる。

次に、ステップ S 174において、CPU 21は、コンテンツに対応して記録するヘッダのデータを作成する。このヘッダのデータは、コンテンツ ID、ライセンス ID、ライセンスを取得するためのアクセス先を表す URL、およびウォーターマークに含まれていたコピー管理情報 (CCI) と、トリガ (Trigger) により構成される。

次に、ステップ S 175に進み、CPU 21は、ステップ S 174の処理で作成したヘッダのデータに基づいたデジタル署名を、自分自身の秘密鍵を用いて作成する。この秘密鍵は、ライセンスサーバ4から取得したものである (図 7 のステップ S 67)。

10 ステップ S 176で、CPU 21は、暗号化復号部 24を制御し、コンテンツキーでコンテンツを暗号化させる。コンテンツキーは、乱数等を用いて生成される。

次に、ステップ S 177において、CPU 21は、ファイルフォーマットに基づき、データを、例えば、ミニディスク等により構成される光磁気ディスク 43 15 に記録させる。

なお、記録媒体がミニディスクである場合、ステップ S 176において、CPU 21は、コンテンツをコーデック部 25に供給し、例えば、ATRAC3方式によりコンテンツを符号化させる。そして、符号化されたデータが暗号化復号部 24によりさらに暗号化される。

20 図 25は、以上のようにして、記録媒体にコンテンツが記録された状態を模式的に表している。暗号化されているコンテンツ (E (At 3)) から抽出されたウォーターマーク (WM) が、コンテンツの外 (ヘッダ) に記録されている。

図 26は、コンテンツを記録媒体に記録する場合のファイルフォーマットのよ 25 り詳細な構成を表している。この例においては、コンテンツ ID (CID) 、ライセンス ID (LID) 、URL、およびウォーターマーク (WM) を含むヘッダが記録されている他、EKB、コンテンツキーKcをルートキーKRで暗号化したデータ (Enc (KR, Kc)) 、証明書 (Cert) 、ヘッダに基づき生成されたデジタル

署名 (Sig (Header) ) 、コンテンツをコンテンツキーKcで暗号化したデータ (Enc (Kc, Content) ) 、メタデータ (Meta Data) およびマーク (Mark) が記録されている。

ウォーターマークは、コンテンツの内部に埋め込まれているものであるが、図 5 25 と図 26 に示されるように、コンテンツの内部とは別に、ヘッダ内に配置するようにすることで、ウォーターマークとしてコンテンツに埋め込まれている情報を迅速に、かつ簡単に検出することが可能となる。従って、そのコンテンツを、コピーすることができるか否かを、迅速に判定することができる。

なお、メタデータは、例えば、ジャケット、写真、歌詞等のデータを表す。マ 10 ークについては、図 3 2 を参照して後述する。

図 2 7 は、証明書としての公開鍵証明書の例を表している。公開鍵証明書は、通常、公開鍵暗号方式における認証局 (CA : Certificate Authority) が発行する証明書であり、ユーザが、認証局に提出した自己の ID や公開鍵などに、認証局が有効期限等の情報を付加し、さらに、認証局によるデジタル署名を付加して 15 作成される。この発明においては、ライセンスサーバ 4 (またはコンテンツサーバ 3) が、証明書と秘密鍵、従って公開鍵も発行するので、ユーザは、ユーザ ID、パスワード等をライセンスサーバ 4 に提供し登録処理を行うことによって、この公開鍵証明書を得ることができる。

図 2 7 における公開鍵証明書は、証明書のバージョン番号、ライセンスサーバ 20 4 が証明書の利用者 (ユーザ) に対して割りつける証明書の通し番号、デジタル署名に用いたアルゴリズム、およびパラメータ、認証局 (ライセンスサーバ 4) の名前、証明書の有効期限、証明書利用者の ID (ノード ID またはリーフ ID) 、並びに証明書利用者の公開鍵が、メッセージとして含まれている。さらに、このメッセージには、認証局としてのライセンスサーバ 4 により作成されたデジタル署名が付加されている。このデジタル署名は、メッセージに対してハッシュ関数を適用して生成されたハッシュ値に基づいて、ライセンスサーバ 4 の秘密鍵を用いて生成されたデータである。

ノード ID またはリーフ ID は、例えば、図 12 の例の場合、デバイス 0 であれば「0 0 0 0」とされ、デバイス 1 であれば「0 0 0 1」とされ、デバイス 15 であれば「1 1 1 1」とされる。このような ID に基づいて、そのデバイス（エンティティ）がツリー構成のどの位置（リーフまたはノード）に位置するエンティティであるのかが識別される。

このように、コンテンツを利用するのに必要なライセンスを、コンテンツとは分離して配布することにより、コンテンツの配布が自由に行われるこことになる。任意の方法、あるいは経路で入手されたコンテンツは、一元的に取り扱うことが可能である。

10 また、ファイルフォーマットを図 26 に示されるように構成することで、そのフォーマットのコンテンツを、インターネットを介して配信する場合は勿論、SDMI (Secure Digital Music Initiative) 機器に提供する場合においても、コンテンツの著作権を管理することが可能となる。

さらに、例えば、図 28 に示されるように、コンテンツが記録媒体を介して提供されたとしても、インターネット 2 を介して提供されたとしても、同様の処理により、SDMI (Secure Digital Music Initiative) 機器としての所定の PD (Portable Device) 等に、チェックアウトしたりすることが可能となる。

次に、図 29 のフローチャートを参照して、クライアント 1 が他のクライアント（例えば、PD）に対してコンテンツをチェックアウトする場合の処理について説明する。

最初に、ステップ S 19 1において、CPU 21 は、コンテンツにデジタル署名が付加されているか否かを判定する。デジタル署名が付加されていると判定された場合、ステップ S 19 2 に進み、CPU 21 は、証明書を抽出し、認証局（ライセンスサーバ 4）の公開鍵で検証する処理を実行する。すなわち、クライアント 1 は、ライセンスサーバ 4 からライセンスサーバ 4 の秘密鍵に対応する公開鍵を取得し、その公開鍵で公開鍵証明書に付加されているデジタル署名を復号する。図 27 を参照して説明したように、デジタル署名は、認証局（ライセンス

サーバ4) の秘密鍵に基づいて生成されており、ライセンスサーバ4の公開鍵を用いて復号することができる。さらに、CPU21は、証明書のメッセージ全体に対してハッシュ関数を適用してハッシュ値を演算する。そしてCPU21は、演算されたハッシュ値と、デジタル署名を復号して得られたハッシュ値とを比較  
5 し、両者が一致すれば、メッセージは改竄されたものではないと判定する。両者が一致しない場合には、この証明書は、改竄されたものであるということになる。

そこで、ステップS193において、CPU21は、証明書が改竄されていないか否かを判定し、改竄されていないと判定された場合、ステップS194に進み、証明書をEKBで検証する処理を実行する。この検証処理は、証明書に含まれるリーフID(図27)に基づいて、EKBをたどることができるか否かを調べることにより行われる。この検証について、図30と図31を参照して説明する。  
10

いま、図30に示されるように、例えば、リーフキーK1001を有するデバイスがリボークされたデバイスであるとする。このとき、図31に示されるようなデータ(暗号化キー)とタグを有するEKBが、各デバイス(リーフ)に配布  
15 される。このEKBは、図30におけるデバイス「1001」をリボークするために、キーKR,K10,K100を更新するEKBとなっている。

リボークデバイス「1001」以外の全てのリーフは、更新されたルートキーK(t)Rを取得することができる。すなわち、ノードキーK0の下位に連なるリーフは、更新されていないノードキーK0を、デバイス内に保持しているので、  
20 暗号化キーEnc(K0, K(t)R)を、キーK0によって復号することで、更新ルートキーK(t)Rを取得することができる。

また、ノード11以下のリーフは、更新されていないノードキーK11を用いて、Enc(K11, K(t)1)をノードキーK11によって復号することで、更新ノードキーK(t)1を取得することができる。さらに、Enc(K(t)1, K(t)R)をノードキーK(t)1によって復号することで、更新ルートキーK(t)Rを取得することが可能となる。ノードキーK101の下位リーフについても、同様に更新ルートキーK(t)Rを取得することが可能である。  
25

さらに、リボーカされていないリーフキーK1000を有するデバイス「1000」は、自己のリーフキーK1000でEnc(K1000, K(t)100)を復号して、ノードキーK(t)100を取得することができ、これを用いてさらに、上位のノードキーを順次復号し、更新ルートキーK(t)Rを取得することができる。

これに対して、リボーカされたデバイス「1001」は、自己のリーフの1段上の更新ノードキーK(t)100を、EKB処理により取得できないので、結局、更新ルートキーK(t)Rを取得することができない。

リボーカされていない正当なデバイス（クライアント1）には、図31に示されるデータとタグを有するEKBが、ライセンスサーバ4から配信され、格納されている。

そこで、各クライアントは、そのタグを利用して、EKB追跡処理を行うことができる。このEKB追跡処理は、上位のルートキーからキー配信ツリーをたどれるか否かを判定する処理である。

15 例えば、図30のリーフ「1001」のID（リーフID）である「1001」を、「1」「0」「0」「1」の4ビットとして把握し、最上位ビットから順次、下位ビットに従って、ツリーをたどることができるか否かが判定される。この判定では、ビットが1であれば、右側に進み、0であれば、左側に進む処理が行われる。

20 ID「1001」の最上位ビットが1であるから、図30のルートキーKRから右側に進む。EKBの最初のタグ（番号0のタグ）は、0:{0, 0}であり、両枝にデータを有するものであると判定される。この場合、右側に進むことができるので、ノードキーK1にたどり着くことができる。

次に、ノードキーK1の下位のノードに進む。ID「1001」の2番目のビットは0であるから左側に進む。番号1のタグは、左側のノードキーK0の下位のデータの有無を表すものであり、ノードキーK1の下位のデータの有無を示すタグは、番号2のタグである。このタグは、図31に示されるように、2:{0,

0} であり、両枝にデータを有するものとされる。従って、左側に進み、ノードキー K 1 0 にたどり着くことができる。

さらに、ID 「1 0 0 1」 の 3 番目のビットは 0 であり、左側に進む。このとき、K 1 0 の下位のデータの有無を示すタグ（番号 3 のタグ）は、3 : { 0 , 5 0 } であり、両枝にデータを有するものと判定される。そこで、左側に進み、ノードキー K 1 0 0 にたどり着くことができる。

さらに、ID 「1 0 0 1」 の最下位ビットは 1 であり、右側に進む。番号 4 のタグは、ノードキー K 1 1 に対応するものであり、K 1 0 0 の下位のデータの符号を表すタグは、番号 5 のタグである。このタグは、5 : { 0 , 1 } である。従 10 って、右側には、データが存在しないことになる。その結果、ノード 「1 0 0 1」 にはたどり着けないことになり、ID 「1 0 0 1」 のデバイスは、EKB による更新ルートキーを取得できないデバイス、すなわちリボーグデバイスであると判定される。

これに対して、例えば、リーフキー K 1 0 0 0 を有するデバイス ID は、「1 15 0 0 0」 であり、上述した場合と同様に、EKB 内のタグに基づく EKB 追跡処理を行うと、ノード 「1 0 0 0」 にたどり着くことができる。従って、ID 「1 0 0 0」 のデバイスは、正当なデバイスであると判定される。

図 2 9 に戻って、CPU 2 1 は、ステップ S 1 9 4 の検証処理に基づき、証明書がリボーグされていないか否かをステップ S 1 9 5 で判定し、証明書がリボー 20 クされていない場合には、ステップ S 1 9 6 に進み、デジタル署名を証明書に含まれる公開鍵で検証する処理を実行する。

すなわち、図 2 7 に示されるように、証明書には、証明書利用者（コンテンツ作成者）の公開鍵が含まれており、この公開鍵を用いて、図 2 6 に示される署名（Sig (Header) ）が検証される。すなわち、この公開鍵を用いて、デジタル 25 署名 Sig (Header) を復号して得られたデータ（ハッシュ値）と、図 2 6 に示される Header にハッシュ関数を適用して演算されたハッシュ値とを比較することで、両者が一致していれば、Header が改竄されていないことを確認するこ

とができる。これに対して、両者が一致しなければ、Header は改竄されているということになる。

ステップ S 197において、CPU 21 は、Header が改竄されているか否かを判定し、改竄されていなければ、ステップ S 198 に進み、ウォーターマーク 5 を検証する。ステップ S 199において、CPU 21 は、ウォーターマークの検証の結果、チェックアウトが可能であるか否かを判定する。チェックアウトが可能である場合には、ステップ S 200 に進み、CPU 21 は、チェックアウトを実行する。すなわち、チェックアウト先のクライアント 1 に対してコンテンツを転送し、コピーさせる。

10   ステップ S 191において、デジタル署名が存在しないと判定された場合、ステップ S 193において、証明書が改竄されていると判定された場合、ステップ S 195において、証明書を EKB で検証することができなかつたと判定された場合、ステップ S 197において、デジタル署名の検証の結果、ヘッダが改竄されていると判定された場合、または、ステップ S 199において、ウォーターマーク 15 にチェックアウトの禁止が記述されていると判定された場合、ステップ S 201 に進み、エラー処理が実行される。すなわち、この場合には、チェックアウトが禁止される。

20   このように、証明書と秘密鍵をライセンスサーバ 4 からユーザに配布し、コンテンツ作成時に、デジタル署名を付加することにより、コンテンツの作成者の真正正を保証することが可能となる。これにより、不正なコンテンツの流通を抑制することができる。

さらに、ウォーターマークをコンテンツ作成時に検出し、その情報をデジタル署名に付することで、ウォーターマーク情報の改竄を防止し、コンテンツの真正を保証することができる。

25   その結果、一度作成されたコンテンツは、どのような形態で配信されたとしても、元のコンテンツの真正を保証することが可能となる。

さらに、コンテンツは、使用条件を有さず、使用条件は、ライセンスに付加されているので、ライセンス内の使用条件を変更することで、それに関するコンテンツの使用条件を一斉に変更することが可能となる。

次に、マークの利用方法について説明する。本発明においては、上述したように、  
5 使用条件は、コンテンツではなく、ライセンスに付加される。しかしながら、  
コンテンツによって、使用状況が異なる場合がある。そこで、本発明においては、  
図 26 に示されるように、コンテンツにマークが付加される。

ライセンスとコンテンツは、1 対多の関係にあるため、コンテンツの個々の使  
用状況をライセンスの使用条件にのみ記述するのは困難となる。そこで、このよ  
10 うに、コンテンツに使用状況を付加することにより、ライセンスでの管理をしな  
がらも、個々のコンテンツを管理することが可能となる。

このマークには、例えば、図 32 に示されるように、ユーザの ID (リーフ  
ID)、所有権フラグ、使用開始時刻、およびコピー回数等が記述される。

さらに、マークには、リーフ ID、所有権フラグ、使用開始時刻、およびコピ  
15 一回数等のメッセージに基づいて生成されたデジタル署名が付加される。

所有権フラグは、例えば、所定の期間だけコンテンツを使用可能とするライセ  
ンスを、そのまま買い取ったような場合（使用期間を永久に変更したような場  
合）に付加される。使用開始時刻は、コンテンツの使用を所定の期間内に開始し  
た場合に記述される。例えば、コンテンツをダウンロードする時期が制限されて  
20 いるような場合において、その期限内にダウンロードが行われたようなとき、そ  
の実際にコンテンツをダウンロードした日時がここに記述される。これにより、  
期間内での有効な使用であることが、証明される。

コピー回数には、それまでにそのコンテンツをコピーした回数が履歴（ログ）  
として記述される。

25 次に、図 33 のフローチャートを参照して、ユーザがライセンスを買い取った  
場合に、マークを付加する処理について、マークをコンテンツに付加する例とし  
て説明する。

最初に、ステップ S 221において、CPU 21は、入力部 26からのユーザの指令に基づいて、インターネット 2を介して、ライセンスサーバ4にアクセスする。

ステップ S 222において、CPU 21は、ユーザからの入力部 26を介して 5 の入力を取り込み、その入力に対応してライセンスサーバ4に対してライセンスの買い取りを要求する。

この要求に対応して、図 34 のフローチャートを参照して後述するように、ライセンスサーバ4は、ライセンスを買い取るために必要な対価を提示していく (図 34 のステップ S 242)。そこで、ステップ S 223において、クライアント 10 1の CPU 21は、ライセンスサーバ4からの対価の提示を受け取ると、これを出力部 27に出力し、表示させる。

ユーザは、この表示に基づいて、提示された対価を了承するか否かを判断し、その判断結果に基づいて、入力部 26からその判断結果を入力する。

CPU 21は、ステップ S 224において、入力部 26からの入力に基づいて、 15 ユーザが提示された対価を了承したか否かを判定し、了承したと判定した場合には、ステップ S 225に進み、ライセンスサーバ4に了承を通知する処理を実行する。

この了承通知を受信すると、ライセンスサーバ4は、対価の買い取りを表す情報、すなわち所有権フラグを記述したマークを送信していく (図 34 のステップ 20 S 244)。そこで、ステップ S 226において、クライアント 1の CPU 21は、ライセンスサーバ4からのマークを受け取ると、ステップ S 227において、受け取ったマークをコンテンツに埋め込む処理を実行する。すなわち、これにより、買い取られたライセンスに対応するコンテンツのマークとして、図 32 に示されるような所有権フラグが記述されたマークがコンテンツに対応して記録されることになる。また、このとき、CPU 21は、メッセージが更新されたことになるので、デジタル署名 (図 26) も更新し、記録媒体に記録する。

ステップ S 224において、ライセンスサーバ4から提示された対価が了承されていないと判定された場合、ステップ S 228に進み、CPU 21は、提示された対価を了承しないことをライセンスサーバ4に通知する。

このようなクライアント1の処理に対応して、ライセンスサーバ4は、図 34 5 のフローチャートに示す処理を実行する。

すなわち、最初に、ステップ S 241において、ライセンスサーバ4の CPU 21は、クライアント1からライセンス買い取りの要求が送信されると（図 33 のステップ S 222）、これを受け取り、ステップ S 242において、対象とされているライセンスの買い取りに必要な対価を記憶部 28 から読み出し、10 これをクライアント1に送信する。

上述したように、このようにして提示された対価に対して、クライアント1から提示された対価を了承するか否かの通知が送信されてくる。

そこで、ステップ S 243において、ライセンスサーバ4の CPU 21は、クライアント1から了承通知を受信したか否かを判定し、了承通知を受信したと判定した場合、ステップ S 244に進み、対象とされるライセンスの買い取りを表すメッセージを含むマークを生成し、自分自身の秘密鍵で、デジタル署名を付加して、クライアント1に送信する。このようにして送信されたマークは、上述したように、クライアント1の記憶部 28 において、対応するコンテンツに記録される（図 33 のステップ S 227）。

20 ステップ S 243において、クライアント1から了承通知が受信されていないと判定された場合、ステップ S 244の処理はスキップされる。すなわち、この場合には、ライセンスの買い取り処理が最終的に行われなかつたことになるので、マークは送信されない。

図 35 は、ステップ S 244において、ライセンスサーバ4からクライアント1に対して送信されるマークの構成例を表している。この例においては、そのユーザのリーフ ID、所有権フラグ（Own）、並びにリーフ ID と所有権フラグを、

ライセンスサーバ4の秘密鍵Sに基づいて生成されたデジタル署名Sigs (LeafID,Own)により、マークが構成されている。

なお、このマークは、特定のユーザの特定のコンテンツに対してのみ有効なものであるので、対象とされるコンテンツがコピーされた場合には、そのコピーされたコンテンツに付随するマークは無効とされる。

このようにして、コンテンツとライセンスを分離し、使用条件をライセンスに対応させる場合においても、個々のコンテンツの使用状況に応じたサービスを実現することが可能となる。

次に、グルーピングについて説明する。複数の機器やメディアを適当に集め、  
10 その1つの集合内においては、コンテンツを自由に授受することができるよう  
にすることは、グルーピングと称される。通常、このグルーピングは、個人の所有  
する機器やメディアにおいて行われる。このグルーピングは、従来、グループ毎  
にグループキーを設定する等して行われていたが、グループ化する複数の機器や  
メディアに、同一のライセンスを対応づけることにより、容易にグルーピングす  
15 ることが可能となる。

また、各機器を予め登録しておくことで、グルーピングすることも可能である。  
この場合のグルーピングについて、以下に説明する。

この場合、ユーザは、グルーピング対象とされる機器の証明書を予めサーバに  
登録しておく必要がある。この証明書の登録処理について、図36と図37のフ  
20 ローチャートを参照して説明する。

最初に、図36のフローチャートを参照して、クライアント（グルーピング対  
象となる機器）の証明書の登録処理について説明する。ステップS261において、クライアント1のCPU21は、グルーピングの対象とされる機器としての  
自己自身の証明書を作成する。この証明書には、自己自身の公開鍵が含まれる。

25 次に、ステップS262に進み、CPU21は、ユーザの入力部26からの入  
力に基づいて、コンテンツサーバ3にアクセスし、ステップS263において、

ステップ S 261 の処理で作成された証明書をコンテンツサーバ 3 に送信する処理を実行する。

なお、証明書としては、ライセンスサーバ 4 から受信したものを、そのまま使用することもできる。

5 以上の処理は、グルーピング対象とされる全ての機器が行う。

次に、図 37 のフローチャートを参照して、図 36 のクライアント 1 の証明書の登録処理に対応して行われるコンテンツサーバ 3 の証明書の登録処理について説明する。

最初に、ステップ S 271において、コンテンツサーバ 3 の CPU 21 は、ク  
10 ライアント 1 から送信されてきた証明書を受信すると、ステップ S 272において、その証明書を記憶部 28 に登録する。

以上の処理が、グループ対象とされる機器毎に行われる。その結果、コンテンツサーバ 3 の記憶部 28 には、例えば、図 38 に示されるように、グループ毎に、そのグループを構成するデバイスの証明書が登録される。

15 図 38 に示される例では、グループ 1 の証明書として、証明書 C11 乃至 C14 が登録されている。これらの証明書 C11 乃至 C14 には、対応する公開鍵 K<sub>P11</sub> 乃至 K<sub>P14</sub> が含まれている。

同様に、グループ 2 の証明書として、証明書 C21 乃至 C23 が登録されており、これらは対応する公開鍵 K<sub>P21</sub> 乃至 K<sub>P23</sub> が含まれている。

20 以上のようなグループを構成する各機器毎に、その証明書が登録された状態において、ユーザからそのグループに属する機器にコンテンツの提供が要求されると、コンテンツサーバ 3 は、図 39 のフローチャートに示す処理を実行する。

最初に、ステップ S 281において、コンテンツサーバ 3 の CPU 21 は、記憶部 28 に記憶されている証明書のうち、そのグループに属する証明書を検証する処理を実行する。

この検証処理は、図 30 と図 31 を参照して説明されたように、各機器の証明書に含まれるリーフ ID に基づいて、タグを利用して EKB をたどることで行わ

れる。EKB は、コンテンツサーバ 3 にも、ライセンスサーバ 4 から配布されている。この検証処理により、リボーカされている証明書は除外される。

ステップ S 282において、コンテンツサーバ 3 の CPU 21 は、ステップ S 281 の検証処理の結果、有効とされた証明書を選択する。そして、ステップ S 283において、CPU 21 は、ステップ S 282 の処理で選択された各機器の証明書の各公開鍵でコンテンツ鍵を暗号化する。ステップ S 284において、CPU 21 は、対象とされるグループの各機器に、ステップ S 283 の処理で暗号化されたコンテンツ鍵をコンテンツとともに送信する。

図 38 に示されるグループ 1 のうち、例えば、証明書 C 14 がリボーカされるとすると、ステップ S 283 の処理で、例えば、図 40 に示されるような暗号化データが生成される。

すなわち、図 40 の例においては、コンテンツ鍵 Kc が、証明書 C 11 の公開鍵 K<sub>P11</sub>、証明書 C 12 の公開鍵 K<sub>P12</sub>、または証明書 C 13 の公開鍵 K<sub>P13</sub>により、暗号化されている。

コンテンツサーバ 3 の図 39 に示されるような処理に対応して、コンテンツの提供を受ける各グループの機器（クライアント）は、図 41 のフローチャートに示す処理を実行する。

最初に、ステップ S 291において、クライアント 1 の CPU 21 は、コンテンツサーバ 3 が図 39 のステップ S 284 の処理で送信してきたコンテンツを、コンテンツ鍵とともに受信する。コンテンツは、コンテンツ鍵 Kc により、暗号化されており、コンテンツ鍵は上述したように、各機器が保持する公開鍵により暗号化されている（図 40）。

そこで、ステップ S 292において、CPU 21 は、ステップ S 291 の処理で受信した自分宛のコンテンツ鍵を、自分自身の秘密鍵で復号し、取得する。そして、取得したコンテンツ鍵を用いてコンテンツの復号処理が行われる。

例えば、図 40 の例に示される証明書 C 11 に対応する機器は、公開鍵 K<sub>P11</sub> に対応する自分自身の秘密鍵を用いて、コンテンツ鍵 Kc の暗号を復号し、コン

テンツ鍵  $K_c$  を取得する。そして、コンテンツ鍵  $K_c$  を用いて、コンテンツがさらに復号される。

同様の処理は、証明書 C 1 2, C 1 3 に対応する機器においても行われる。リボーカされている証明書 C 1 4 の機器は、自分自身の公開鍵を用いて暗号化されたコンテンツ鍵  $K_c$  がコンテンツに付随して送られてこないので、コンテンツ鍵  $K_c$  を復号することができず、従って、コンテンツ鍵  $K_c$  を用いてコンテンツを復号することができない。

以上においては、コンテンツキー（すなわちコンテンツ）に対してグルーピングを行うようにしたが、ライセンスキー（ライセンス）に対してグルーピングを 10 行うことも可能である。

以上のようにして、特別なグループキー、後述する ICV (Integrity Check Value) を用いずにグループ化が可能となる。このグループ化は、小規模のグループに適用するのに向いている。

本発明においては、ライセンスもチェックアウト、あるいはチェックインしたり、ムーブしたり、コピーしたりすることが可能とされる。但し、これらの処理は SDMI で定められたルールに基づいて行われる。

次に、図 4 2 と図 4 3 のフローチャートを参照して、このようなクライアントによるライセンスのチェックアウト処理について説明する。

最初に、図 4 2 のフローチャートを参照して他のクライアントにライセンスを 20 チェックアウトするクライアントの処理について説明する。最初に、ステップ S 3 0 1において、クライアント 1 の CPU 2 1 は、チェックアウト対象のライセンスのチェックアウト回数  $N_1$  を読み取る。このチェックアウト回数は、図 8 に示される使用条件に書き込まれているので、この使用条件から読み取られる。

次に、ステップ S 3 0 2において、CPU 2 1 は、チェックアウト対象のライセンスの最大チェックアウト回数  $N_2$  を、やはりライセンスの使用条件から読み取る。

そして、ステップ S 303において、CPU21は、ステップ S 301の処理で読み取られたチェックアウト回数N1と、ステップ S 302の処理で読み取られた最大チェックアウト回数N2とを比較し、チェックアウト回数N1が最大チェックアウト回数N2より小さいか否かを判定する。

- 5 チェックアウト回数N1が、最大チェックアウト回数N2より小さいと判定された場合、ステップ S 304に進み、CPU21は、相手側の装置（チェックアウト先のクライアント）のリーフキーを相手個々の装置から取得し、そのリーフキーを、いまチェックアウト対象とされているライセンスIDに対応して記憶部28のチェックアウトリストに記憶させる。
- 10 次に、ステップ S 305において、CPU21は、ステップ S 301の処理で読み取られたライセンスのチェックアウト回数N1の値を1だけインクリメントする。ステップ S 306において、CPU21は、ライセンスのメッセージに基づいて、ICVを演算する。このICVについては、図47乃至図51を参照して後述する。ICVを用いてライセンスの改竄を防止することが可能となる。
- 15 次に、ステップ S 307において、CPU21は、チェックアウト対象のライセンスと、ステップ S 306の処理で演算されたICVを、自分自身の公開鍵を用いて暗号化して、EKBおよび証明書とともに、相手側の装置に出力し、コピーさせる。さらに、ステップ S 308において、CPU21は、ステップ S 306の処理で演算されたICVを、相手側装置のリーフキーと、ライセンスIDに対応して記憶部28のチェックリスト中に記憶させる。
- 20

ステップ S 303において、チェックアウト回数N1が最大チェックアウト回数N2より小さくない（例えば、等しい）と判定された場合、もはや許容される回数だけチェックアウトが行われているので、これ以上チェックアウトを行うことができない。そこで、ステップ S 309に進み、CPU21は、エラー処理を実行する。すなわち、この場合、チェックアウト処理は実行されないことになる。

次に、図43のフローチャートを参照して、図42のチェックアウト処理により、ライセンスのチェックアウトを受けるクライアントの処理について説明する。

最初に、ステップ S 321において、相手側装置（ライセンスをチェックアウトするクライアント1）に、自分自身のリーフキーを送信する。このリーフキーは、ステップ S 304において、相手側のクライアントにより、ライセンス ID に対応して記憶される。

- 5 次に、ステップ S 322において、CPU21は、相手側のクライアント1から暗号化されたライセンスとICVが、EKB および証明書とともに送信されてきた場合、これを受信する。すなわち、このライセンス、ICV、EKB および証明書は、図42のステップ S 307の処理で相手側の装置から送信されたものである。
- 10 ステップ S 323において、CPU21は、ステップ S 322の処理で受信したライセンス、ICV、EKB および証明書を、記憶部28に記憶させる。

以上のようにして、ライセンスのチェックアウトを受けたクライアント1は、チェックアウトを受けたそのライセンスを使用して、所定のコンテンツを再生する場合、図44のフローチャートに示される処理を実行する。
- 15 すなわち、最初に、ステップ S 341において、クライアント1のCPU21は、ユーザより入力部26を介して再生が指定されたコンテンツのICVを演算する。そして、ステップ S 342において、CPU21は、記憶部28に記憶されている暗号化されているICVを、証明書に含まれている公開鍵に基づいて、復号させる。
- 20 次に、ステップ S 343において、CPU21は、ステップ S 341の処理により、いま演算されたICVと、ステップ S 342の処理により読み出され、復号されたICVが一致するか否かを判定する。両者が一致する場合には、ライセンスは改竄されていないことになる。そこで、ステップ S 344にすすみ、CPU21は、対応するコンテンツを再生する処理を実行する。
- 25 これに対して、ステップ S 343において、2つのICVが一致しないと判定された場合、ライセンスは改竄されている恐れがある。このため、ステップ S 3

45に進み、CPU21は、エラー処理を実行する。すなわち、このとき、そのライセンスを用いてコンテンツを再生することができないことになる。

次に、以上のようにして、他のクライアントに一旦チェックアウトしたライセンスのチェックインを受けるクライアントの処理について、図45のフローチャ

5 ートを参照して説明する。

最初に、ステップS361において、CPU21は、相手側の装置（ライセンスを返却（チェックイン）してくるクライアント1）のリーフキーと、チェックイン対象のライセンスのIDを取得する。次に、ステップS362において、

CPU21は、ステップS361で取得されたチェックイン対象のライセンスが、

10 自分自身が相手側装置にチェックアウトしたライセンスであるか否かを判定する。

この判定は、図42のステップS308の処理で記憶されたICV、リーフキー、およびライセンスIDに基づいて行われる。すなわち、ステップS361で取得されたリーフキー、ライセンスID、およびICVが、チェックアウトリスト中に記憶されているか否かが判定され、記憶されている場合には、自分自身がチエッ

15 クアウトしたライセンスであると判定される。

ライセンスが、自分自身がチェックアウトしたものであるとき、ステップS363において、CPU21は、相手側の装置のライセンス、EKBおよび証明書の削除を要求する。後述するように、この要求に基づいて、相手側の装置は、ライセンス、EKBおよび証明書の削除を実行する（図46のステップS383）。

20 ステップS364において、CPU21は、一旦チェックアウトしたライセンスが再びチェックインされてきたので、そのライセンスのチェックアウト回数N1を1だけデクリメントする。

ステップS365において、CPU21は、相手側の装置に他のライセンスをチェックアウトしているか否かを判定し、まだチェックアウトしている他のライ

25 センスが存在しない場合には、ステップS366に進み、CPU21は、相手側の装置のチェックイン対象機器としてのチェックアウトリストにおける記憶を削除する。これに対して、ステップS365において、相手側の装置にチェックア

ウトしている他のライセンスが存在すると判定された場合には、他のライセンスのチェックインを受ける可能性があるので、ステップ S 3 6 6 の処理はスキップされる。

ステップ S 3 6 2 において、チェックイン対象とされているライセンスが、自分自身が相手側装置にチェックアウトしたライセンスではないと判定された場合、CPU 2 1 は、ステップ S 3 6 7 に進み、エラー処理を実行する。すなわち、この場合には、自分自身が管轄するライセンスではないことになるので、チェックイン処理は実行されない。

ユーザが、ライセンスを不正にコピーしたような場合、記憶されている ICV の値と、ステップ S 3 6 1 の処理で取得されたライセンスに基づいて演算された ICV の値が異なるものとなるで、チェックインできないことになる。

図 4 6 は、図 4 5 のフローチャートに示されるライセンスのチェックイン処理を実行するクライアントに対して、自分自身が有しているライセンスをチェックインさせるクライアントの処理を表している。

ステップ S 3 8 1 において、クライアント 1 の CPU 2 1 は、相手側の装置（図 4 5 のフローチャートに示す処理を実行するクライアント 1）にリーフキーとチェックイン対象のライセンスの ID を送信する。上述したように、相手側の装置は、ステップ S 3 6 1 において、このリーフキーとライセンス ID を取得し、ステップ S 3 6 2 において、それに基づいて、チェックイン対象のライセンスの認証処理を実行する。

ステップ S 3 8 2 において、クライアント 1 の CPU 2 1 は、相手側の装置からライセンスの削除を要求されたか否かを判定する。すなわち、ライセンスが正当なチェックイン対象のライセンスである場合、上述したように、相手側の装置は、ステップ S 3 6 3 の処理でライセンス、EKB および証明書の削除を要求してくれる。そこで、この要求を受信した場合、ステップ S 3 8 3 に進み、CPU 2 1 は、ライセンス、EKB および証明書を削除する。すなわち、これにより、このクライアント 1 は、以後そのライセンスを使用できない状態となり、図 4 5 の

ステップS364の処理により、チェックアウト回数N1が、1だけデクリメントされるので、チェックインが完了したことになる。

ステップS382において、相手側の装置からライセンスの削除が要求されていないと判定された場合、ステップS384に進み、エラー処理が実行される。

5 すなわち、この場合には、ICVの値が異なっている等の理由により、チェックインができないことになる。

以上においては、チェックインとチェックアウトについて説明したが、同様に、ライセンスをコピーあるいはムーブさせるようにすることも可能である。

次に、ライセンス（コンテンツも同様）の改竄を防止するためにライセンスの  
10 インテグリティ・チェック値（ICV）を生成して、ライセンスに対応付けて、  
ICVの計算により、ライセンス改竄の有無を判定する処理構成について説明す  
る。

ライセンスのインテグリティ・チェック値（ICV）は、例えばライセンスに  
15 対するハッシュ関数を用いて計算され、 $ICV = \text{hash}(K_{icv}, L_1, L_2, \dots)$  によって計算される。 $K_{icv}$  は ICV 生成キーである。 $L_1, L_2$  はライセンスの情報であり、ライセンスの重要な情報のメッセージ認証符号（  
MAC : Message authentication Code）が使用される。

DES 暗号処理構成を用いた MAC 値生成例を図 4-7 に示す。図 4-7 の構成に  
示すように対象となるメッセージを 8 バイト単位に分割し、（以下、分割された  
20 メッセージを  $M_1, M_2, \dots, M_N$  とする）、まず、初期値（IV）と  $M_1$   
を、演算部 24-1A により排他的論理和する（その結果を  $I_1$  とする）。次に、  
 $I_1$  を DES 暗号化部 24-1B に入れ、鍵（以下、 $K_1$  とする）を用いて暗号  
化する（出力を  $E_1$  とする）。続けて、 $E_1$  および  $M_2$  を演算部 24-2A によ  
り排他的論理和し、その出力  $I_2$  を DES 暗号化部 24-2B に入れ、鍵  $K_1$  を  
25 用いて暗号化する（出力  $E_2$ ）。以下、これを繰り返し、全てのメッセージに  
して暗号化処理を施す。DES 暗号化部 24-NB から最後に出てきた  $E_N$  がメ  
ッセージ認証符号（MAC (Message Authentication Code)）となる。

このようなライセンスのMAC値とICV生成キーにハッシュ関数を適用してライセンスのインテグリティ・チェック値(ICV)が生成される。例えばライセンス生成時に生成したICVと、新たにライセンスに基づいて生成したICVとを比較して同一のICVが得られればライセンスに改竄のないことが保証され、

5 ICVが異なれば、改竄があったと判定される。

次に、ライセンスのインテグリティ・チェック値(ICV)生成キーである $K_{icv}$ を上述の有効化キーブロックによって送付する構成について説明する。すなわちEKBによる暗号化メッセージデータをライセンスのインテグリティ・チェック値(ICV)生成キーとした例である。

10 図48および図49に複数のデバイスに共通のライセンスを送付した場合、それらのライセンスの改竄の有無を検証するためのインテグリティ・チェック値生成キー $K_{icv}$ を有効化キーブロック(EKB)によって配信する構成例を示す。図48はデバイス0, 1, 2, 3に対して復号可能なチェック値生成キー $K_{icv}$ を配信する例を示し、図49はデバイス0, 1, 2, 3中のデバイス3をリボーグ(排除)してデバイス0, 1, 2に対してのみ復号可能なチェック値生成キー $K_{icv}$ を配信する例を示す。

15

図48の例では、更新ノードキー $K(t)_{00}$ によって、チェック値生成キー $K_{icv}$ を暗号化したデータ $E_{nc}(K(t)_{00}, K_{icv})$ とともに、デバイス0, 1, 2, 3においてそれぞれの有するノードキー、リーフキーを用いて更新されたノードキー $K(t)_{00}$ を復号可能な有効化キーブロック(EKB)を生成して配信する。それぞれのデバイスは、図48の右側に示すように、まず、EKBを処理(復号)することにより、更新されたノードキー $K(t)_{00}$ を取得し、次に、取得したノードキー $K(t)_{00}$ を用いて、暗号化されたチェック値生成キー $E_{nc}(K(t)_{00}, K_{icv})$ を復号して、チェック値生成キー $K_{icv}$ を得ることが可能となる。

他のデバイス4, 5, 6, 7...は同一の有効化キーブロック(EKB)を受信しても自身の保有するノードキー、リーフキーでは、EKBを処理し

て更新されたノードキー  $K(t)_{00}$  を取得することができないので、安全に正当なデバイスに対してのみチェック値生成キーを送付することができる。

一方、図 4 9 の例は、図 1 2 の点線枠で囲んだグループにおいてデバイス 3 が、例えば鍵の漏洩によりリボーカ（排除）されているとして、他のグループのメンバ、すなわち、デバイス 0, 1, 2, に対してのみ復号可能な有効化キーブロック（EKB）を生成して配信した例である。図 4 9 に示す有効化キーブロック（EKB）と、チェック値生成キー（Kicv）をノードキー（ $K(t)_{00}$ ）で暗号化したデータ  $E_{nc}(K(t)_{00}, Kicv)$  を配信する。

図 4 9 の右側には、復号手順を示してある。デバイス 0, 1, 2 は、まず、受領した有効化キーブロックから自身の保有するリーフキーまたはノードキーを用いた復号処理により、更新ノードキー（ $K(t)_{00}$ ）を取得する。次に、 $K(t)_{00}$  による復号によりチェック値生成キー  $Kicv$  を取得する。

図 1 2 に示す他のグループのデバイス 4, 5, 6, は、この同様のデータ（EKB）を受信したとしても、自身の保有するリーフキー、ノードキーを用いて更新ノードキー（ $K(t)_{00}$ ）を取得することができない。同様にリボーカされたデバイス 3 においても、自身の保有するリーフキー、ノードキーでは、更新ノードキー（ $K(t)_{00}$ ）を取得することができず、正当な権利を有するデバイスのみがチェック値生成キーを復号して利用することが可能となる。

このように、EKB を利用したチェック値生成キーの配信を用いれば、データ量を少なくして、かつ安全に正当権利者のみが復号可能としたチェック値生成キーを配信することが可能となる。

このようなライセンスのインテグリティ・チェック値（ICV）を用いることにより、EKB と暗号化ライセンスの不正コピーを排除することができる。例えば図 5 0 A に示すように、ライセンス L1 とライセンス L2 とをそれぞれのライセンスキーを取得可能な有効化キーブロック（EKB）とともに格納したメディア 1 があり、これをそのままメディア 2 にコピーした場合を想定する。EKB と

暗号化ライセンスのコピーは可能であり、これを、EKBを復号可能なデバイスでは利用できることになる。

図50Bに示す例では、各メディアに正当に格納されたライセンスに対応付けてインテグリティ・チェック値（ICV（L1, L2））を格納する構成とする。

5 なお、（ICV（L1, L2））は、ライセンスL1とライセンスL2にハッシュ関数を用いて計算されるライセンスのインテグリティ・チェック値であるICV = hash（Kicv, L1, L2）を示している。図50Bの構成において、メディア1には正当にライセンス1とライセンス2が格納され、ライセンスL1とライセンスL2に基づいて生成されたインテグリティ・チェック値（ICV（L1, L2））が格納される。また、メディア2には正当にライセンス1が格納され、ライセンスL1に基づいて生成されたインテグリティ・チェック値（ICV（L1））が格納される。

この構成において、メディア1に格納された（EKB, ライセンス2）をメディア2にコピーしたとすると、メディア2で、ライセンスチェック値を新たに生成すると、ICV（L1, L2）が生成されることになり、メディア2に格納されているKicv（L1）と異なり、ライセンスの改竄あるいは不正なコピーによる新たなライセンスの格納が実行されたことが明らかになる。メディアを再生するデバイスにおいて、再生ステップの前ステップにICVチェックを実行して、生成ICVと格納ICVの一致を判別し、一致しない場合は、再生を実行しない構成とすることにより、不正コピーのライセンスの再生を防止することが可能となる。

また、さらに、安全性を高めるため、ライセンスのインテグリティ・チェック値（ICV）を書き換えカウンタを含めたデータに基づいて生成する構成としてもよい。すなわちICV = hash（Kicv, counter + 1, L1, L2, . . .）によって計算する構成とする。ここで、カウンタ（counter + 1）は、ICVの書き換えごとに1つインクリメントされる値として設定する。なお、カウンタ値はセキュアなメモリに格納する構成とする必要である。

さらに、ライセンスのインテグリティ・チェック値（ICV）をライセンスと同一メディアに格納することができない構成においては、ライセンスのインテグリティ・チェック値（ICV）をライセンスとは別のメディア上に格納する構成としてもよい。

5 例えば、読み込み専用メディアや通常のMO等のコピー防止策のとられていないメディアにライセンスを格納する場合、同一メディアにインテグリティ・チェック値（ICV）を格納するとICVの書き換えが不正なユーザによりなされる可能性があり、ICVの安全性が保てないおそれがある。この様な場合、ホストマシン上の安全なメディアにICVを格納して、ライセンスのコピーコントロール（例えばcheck-in/check-out、move）にICVを使用する構成とすることにより、ICVの安全な管理およびライセンスの改竄チェックが可能となる。

10 この構成例を図51に示す。図51では読み込み専用メディアや通常のMO等のコピー防止策のとられていないメディア2201にライセンス1乃至ライセンス3が格納され、これらのライセンスに関するインテグリティ・チェック値（ICV）を、ユーザが自由にアクセスすることの許可されないホストマシン上の安全なメディア2202に格納し、ユーザによる不正なインテグリティ・チェック値（ICV）の書き換えを防止した例である。このような構成として、例えばメディア2201を装着したデバイスが、メディア2201の再生を実行する際にホストマシンであるPC、サーバにおいてICVのチェックを実行して再生の可20能を判定する構成とすれば、不正なコピーライセンスあるいは改竄ライセンスの再生を防止できる。

本発明が適用されるクライアントは、いわゆるパーソナルコンピュータ以外に、PDA（Personal Digital Assistants）、携帯電話機、ゲーム端末機などとすることができる。

25 一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行すること

が可能な、例えば汎用のパーソナルコンピュータなどに、ネットワークや記録媒体からインストールされる。

この記録媒体は、図2に示されるように、装置本体とは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク41（フロッピディスクを含む）、光ディスク42（CD-ROM（Compact Disk - Read Only Memory）、DVD（Digital Versatile Disk）を含む）、光磁気ディスク43（MD（Mini-Disk）を含む）、もしくは半導体メモリ44などよりなるパッケージメディアにより構成されるだけでなく、装置本体に予め組み込まれた状態でユーザに提供される、プログラムが記録されているROM22や、記憶部28に含まれるハードディスクなどで構成される。

なお、本明細書において、記録媒体に記録されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理はもちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

また、セキュリティに関連する処理を実行させるプログラムは、その処理を解析されるのを防ぐため、そのプログラム自体が暗号化されているのが望ましい。例えば、暗号処理などを行う処理については、そのプログラムをタンパーレジスタントモジュールとして構成することができる。

また、コンテンツを利用許可するライセンスを特定するためにコンテンツのヘッダに記載されている情報はライセンスを一意に識別するライセンスIDでなくてもよい。上記の実施例では、ライセンスIDが、コンテンツの利用に必要なライセンスを特定する情報であり、あるライセンスが利用を許可するコンテンツを特定する情報であり、クライアント1からライセンス要求によって要求されるライセンスを識別する情報である。コンテンツにコンテンツのそのコンテンツに関する各種属性情報のリストが記載され、ライセンスに、そのライセンスによって利用許可されるコンテンツの条件式を記載するようにしても良い。この場合には、コンテンツに含まれる属性情報がそのコンテンツの利用を許可するライセンスを

特定する情報であり、ライセンスに含まれる条件式がそのライセンスが利用を許可するコンテンツを特定する情報であり、ライセンス ID はライセンスを一意に識別する情報となる。このようにした場合には、一つのコンテンツに複数のライセンスを対応付けることが可能になり、ライセンスの発行を柔軟に行うことができる。

また、コンテンツデータは音楽データに限らない。例えばコンテンツは、画像データ、動画データ、テキストデータ、アニメーションデータ、ソフトウェアプログラム、あるいはそれらを組み合わせたものであっても良い。

また、本明細書において、システムとは、複数の装置により構成される装置全体を表すものである。

#### 産業上の利用可能性

以上の如く、本発明の第 1 の情報処理装置によれば、供与形態毎に鍵を管理することができる。

本発明の第 2 の情報処理装置によれば、供与形態毎に鍵を管理することができる。

本発明の第 3 の情報処理装置によれば、複数のコンテンツの供与形態に対応することができる。

本発明の第 4 の情報処理装置によれば、複数の異なるデバイスノードキーを保持することができる。

## 請求の範囲

1. コンテンツに含まれる有効化キーブロック (EKB) の復号処理に使われるデバイスノードキーを提供する情報処理装置であって、  
他の情報処理装置を、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、前記第1のノードの下位にあるリーフに、一意に割り当てる割り当て手段と、  
前記割り当て手段により割り当てられた前記リーフから前記第1のノードまでのパスに対応するデバイスノードキーを、前記他の情報処理装置に提供する提供手段と  
10 を備えることを特徴とする情報処理装置。
2. 前記デバイスノードキーと共にリーフを識別するリーフ識別情報を提供することを特徴とする請求の範囲第1項に記載の情報処理装置。
3. 前記他の情報処理装置から前記リーフ識別情報とコンテンツの利用を許可するためのライセンスを識別するライセンス識別情報とを含むライセンス要求を受信する受信手段と、  
前記ライセンス要求に含まれるリーフ識別情報が付加され電子署名がなされたライセンスを送信する送信手段と  
をさらに備えることを特徴とする請求の範囲第2項に記載の情報処理装置。  
20 4. 前記デバイスノードキーと共に、前記他の情報処理装置の秘密鍵および公開鍵、並びに自分自身の公開鍵を提供する  
ことを特徴とする請求の範囲第1項に記載の情報処理装置。
5. コンテンツに含まれる有効化キーブロック (EKB) の復号処理に使われるデバイスノードキーを提供する情報処理方法であって、  
25 他の情報処理装置を、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、前記第1のノードの下位にあるリーフに、一意に割り当てる割り当てステップと、

前記割り当てステップの処理により割り当てられた前記リーフから前記第1のノードまでのパスに対応するデバイスノードキーを、前記他の情報処理装置に提供する提供ステップと

を含むことを特徴とする情報処理方法。

5 6. コンテンツに含まれる有効化キーブロック (EKB) の復号処理に使われるデバイスノードキーを提供する情報処理装置用のプログラムであって、他の情報処理装置を、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、前記第1のノードの下位にあるリーフに、一意に割り当てる割り当てステップと、

10 前記割り当てステップの処理により割り当てられた前記リーフから前記第1のノードまでのパスに対応するデバイスノードキーを、前記他の情報処理装置に提供する提供ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

15 7. コンテンツに含まれる有効化キーブロック (EKB) の復号処理に使われるデバイスノードキーを提供する情報処理装置を制御するコンピュータが実行可能なプログラムであって、他の情報処理装置を、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、前記第1のノードの下位にあるリーフに、一意に割り当てる割り当てステップと、

前記割り当てステップの処理により割り当てられた前記リーフから前記第1のノードまでのパスに対応するデバイスノードキーを、前記他の情報処理装置に提供する提供ステップと

を含むことを特徴とするプログラム。

25 8. コンテンツを利用するためのライセンスを提供する情報処理装置であって、他の情報処理装置から供給された、前記他の情報処理装置に割り当てられた鍵管理階層ツリー構造のリーフを識別するリーフ識別情報を受信する受信手段と、

前記受信手段により受信された前記リーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、前記第1のノードの下位にあるリーフのリーフ識別情報である場合に、前記リーフ識別情報を含むライセンスを送信する送信手段と

5 を備えることを特徴とする情報処理装置。

9. コンテンツを利用するためのライセンスを提供する情報処理方法であって、他の情報処理装置から供給された、前記他の情報処理装置に割り当てられた鍵管理課位相ツリー構造のリーフを識別するリーフ識別情報を受信する受信ステップと、

10 前記受信ステップの処理により受信された前記リーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、前記第1のノードの下位にあるリーフのリーフ識別情報である場合に、前記リーフ識別情報を含むライセンスを送信する送信ステップとを含むことを特徴とする情報処理方法。

15 10. コンテンツを利用するためのライセンスを提供する情報処理装置用のプログラムであって、

他の情報処理装置から供給された、前記他の情報処理装置に割り当てられた鍵管理課位相ツリー構造のリーフを識別するリーフ識別情報を受信する受信ステップと、

20 前記受信ステップの処理により受信された前記リーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、前記第1のノードの下位にあるリーフのリーフ識別情報である場合に、前記リーフ識別情報を含むライセンスを送信する送信ステップとを含むことを特徴とするコンピュータが読み取り可能なプログラムが記録され

25 ている記録媒体。

11. コンテンツを利用するためのライセンスを提供する情報処理装置を制御するコンピュータが実行可能なプログラムであって、

他の情報処理装置から供給された、前記他の情報処理装置に割り当てられた鍵管理課位相ツリー構造のリーフを識別するリーフ識別情報を受信する受信ステップと、

前記受信ステップの処理により受信された前記リーフ識別情報が、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における、前記第1のノードの下位にあるリーフのリーフ識別情報である場合に、前記リーフ識別情報を含むライセンスを送信する送信ステップとを含むことを特徴とするプログラム。

1 2. コンテンツを提供する情報処理装置であって、

コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求を受信する受信手段と、

所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における前記第1のノードの下位にあるリーフから前記第1のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック (EKB) を含む暗号化コンテンツを送信する送信手段とを備えることを特徴とする情報処理装置。

1 3. コンテンツを提供する情報処理方法であって、

コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求を受信する受信ステップと、

所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における前記第1のノードの下位にあるリーフから前記第1のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック (EKB) を含む暗号化コンテンツを送信する送信ステップとを含むことを特徴とする情報処理方法。

2 5. 1 4. コンテンツを提供する情報処理装置用のプログラムであって、

コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求を受信する受信ステップと、

所定の階層の第 1 のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における前記第 1 のノードの下位にあるリーフから前記第 1 のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック (EKB) を含む暗号化コンテンツを送信する送信ステップと

5 を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

15. コンテンツを提供する情報処理装置を制御するコンピュータが実行可能なプログラムであって、

コンテンツを識別するコンテンツ識別情報を含むコンテンツ要求を受信する受  
10 信ステップと、

所定の階層の第 1 のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における前記第 1 のノードの下位にあるリーフから前記第 1 のノードまでのパスに対応するデバイスノードキーによって復号処理可能な有効化キーブロック (EKB) を含む暗号化コンテンツを送信する送信ステップと

15 を含むことを特徴とするプログラム。

16. コンテンツを出力する情報処理装置であって、

自分自身に割り当てられた、所定の階層の第 1 のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における前記第 1 のノードの下位にあるリーフから、前記第 1 のノードまでに対応するデバイスノードキーを記  
20 憶する記憶手段と、

前記第 1 のノードをルートキーに対応させた有効化キーブロック (EKB) を含む暗号化コンテンツを取得するコンテンツ取得手段と、

前記コンテンツ取得手段により取得された前記暗号化コンテンツに含まれる有効化キーブロック (EKB) を前記記憶手段に記憶されているデバイスノードキーを用いて復号処理することで、前記暗号化コンテンツを復号する復号手段と、

前記復号手段により復号されたコンテンツを出力する出力手段と  
を備えることを特徴とする情報処理装置。

17. コンテンツを出力する情報処理方法であって、

自分自身に割り当てられた、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における前記第1のノードの下位にあるリーフから、前記第1のノードまでに対応するデバイスノードキーの記憶を制御する記憶制御ステップと、

前記第1のノードをルートキーに対応させた有効化キーブロック (EKB) を含む暗号化コンテンツを取得するコンテンツ取得ステップと、

前記コンテンツ取得ステップの処理により取得されたコンテンツに含まれる有効化キーブロック (EKB) を前記記憶制御ステップの処理により記憶が制御さ

10 れているデバイスノードキーを用いて復号処理することで、前記暗号化コンテンツを復号する復号ステップと、

前記復号ステップの処理により復号されたコンテンツを出力する出力ステップと

を含むことを特徴とする情報処理方法。

15 18. コンテンツを出力する情報処理装置用のプログラムであって、

自分自身に割り当てられた、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における前記第1のノードの下位にあるリーフから、前記第1のノードまでに対応するデバイスノードキーの記憶を制御する記憶制御ステップと、

20 前記第1のノードをルートキーに対応させた有効化キーブロック (EKB) を含む暗号化コンテンツを取得するコンテンツ取得ステップと、

前記コンテンツ取得ステップの処理により取得されたコンテンツに含まれる有効化キーブロック (EKB) を前記記憶制御ステップの処理により記憶が制御さ  
れていますデバイスノードキーを用いて復号処理することで、前記暗号化コンテン

25 ツを復号する復号ステップと、

前記復号ステップの処理により復号されたコンテンツを出力する出力ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

19. コンテンツを出力する情報処理装置を制御するコンピュータが実行可能なプログラムであって、

5 自分自身に割り当てられた、所定の階層の第1のノードにコンテンツの供与形態の分類が割り当てられた鍵管理階層ツリー構造における前記第1のノードの下位にあるリーフから、前記第1のノードまでに対応するデバイスノードキーの記憶を制御する記憶制御ステップと、

前記第1のノードをルートキーに対応させた有効化キーブロック (EKB) を  
10 含む暗号化コンテンツを取得するコンテンツ取得ステップと、

前記コンテンツ取得ステップの処理により取得されたコンテンツに含まれる有効化キーブロック (EKB) を前記記憶制御ステップの処理により記憶が制御されているデバイスノードキーを用いて復号処理することで、前記暗号化コンテンツを復号する復号ステップと、

15 前記復号ステップの処理により復号されたコンテンツを出力する出力ステップ  
と

を含むことを特徴とするプログラム。

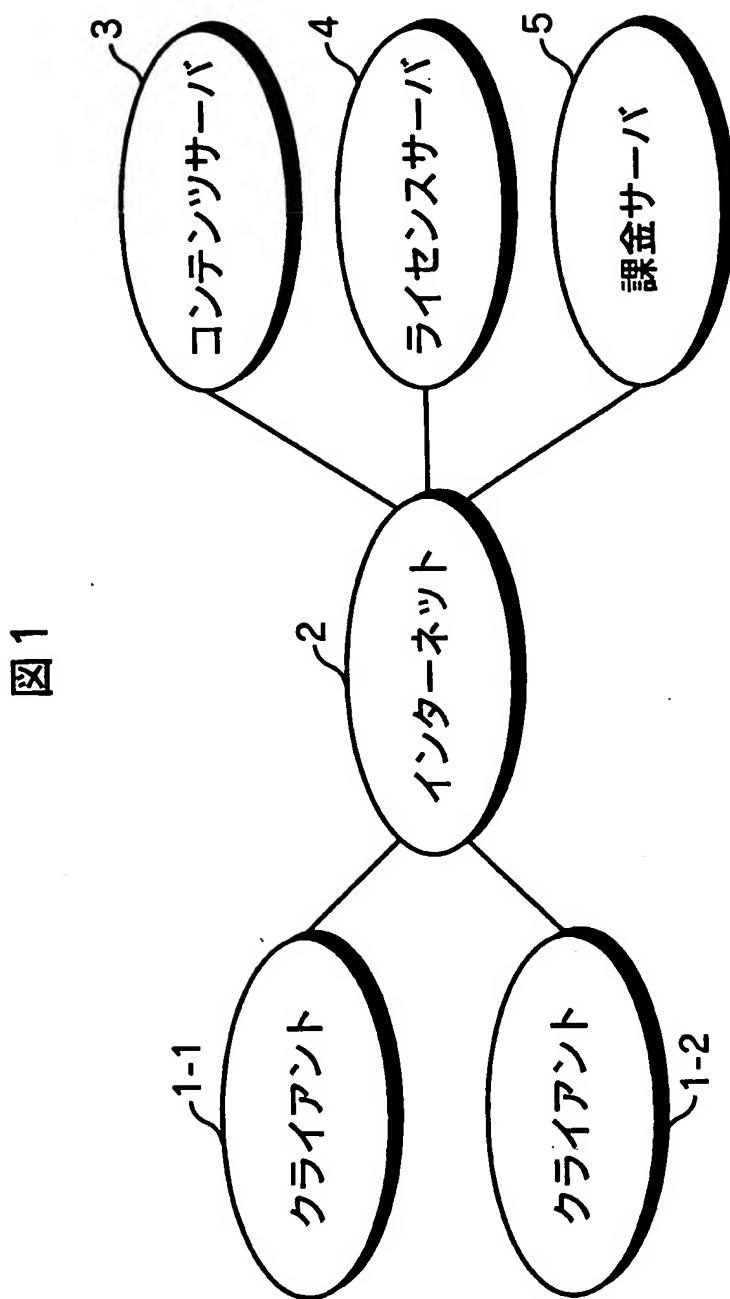


図2

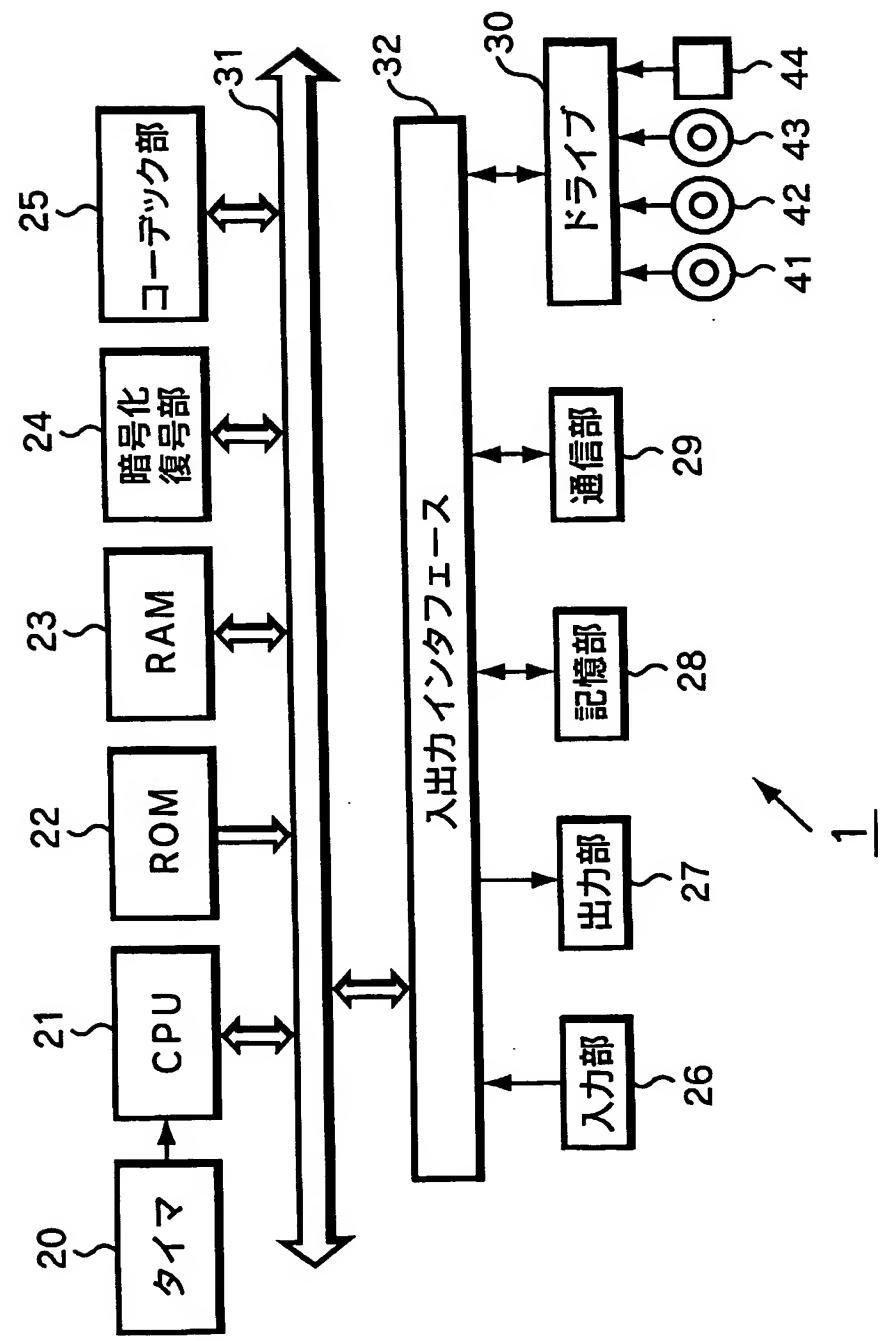
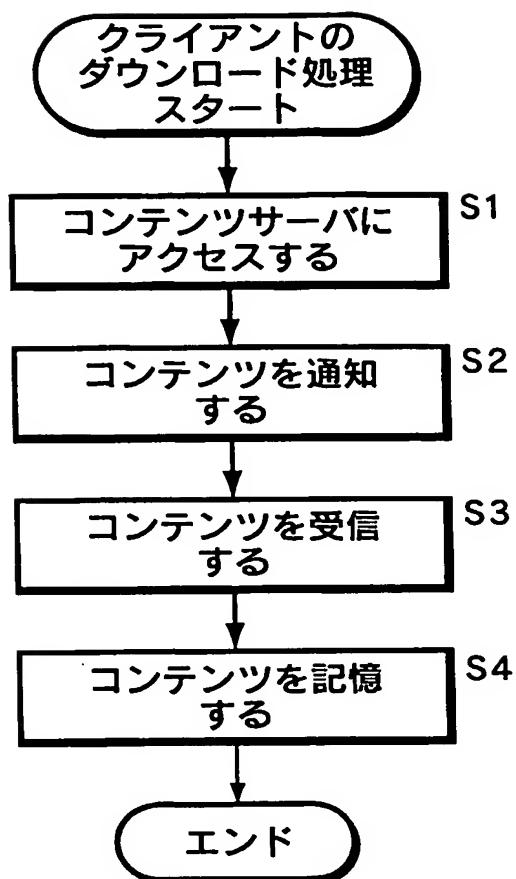


図3



4/45

図 4

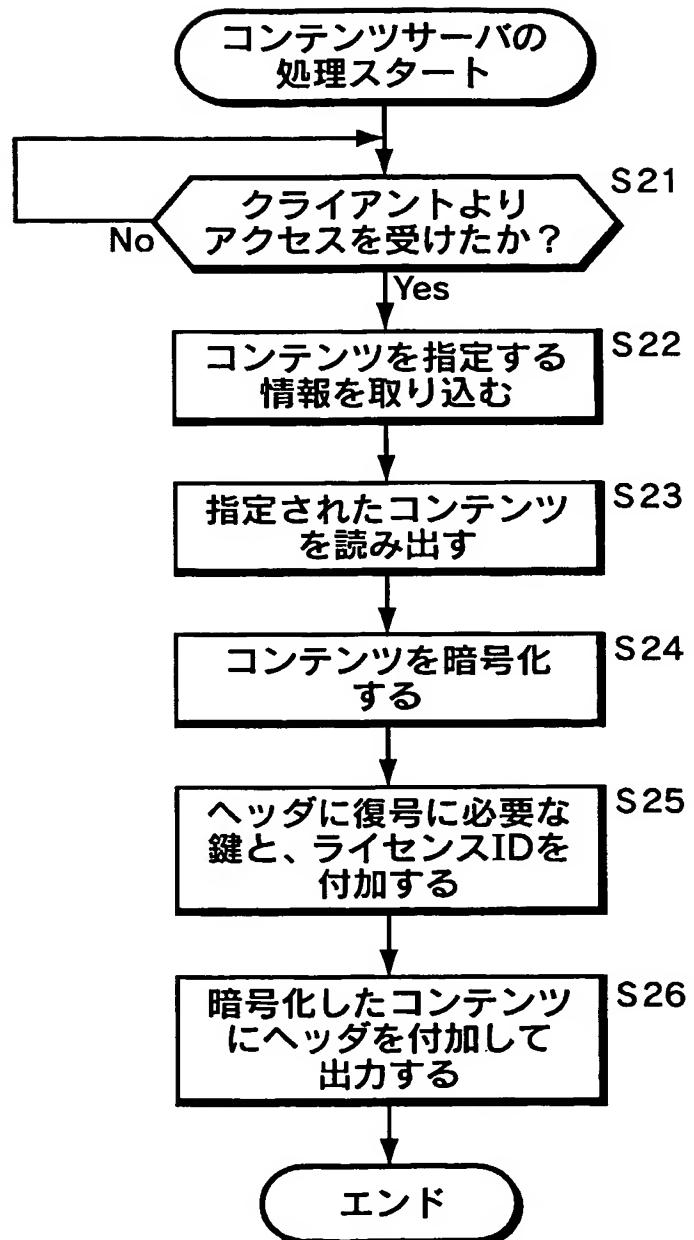


図 5

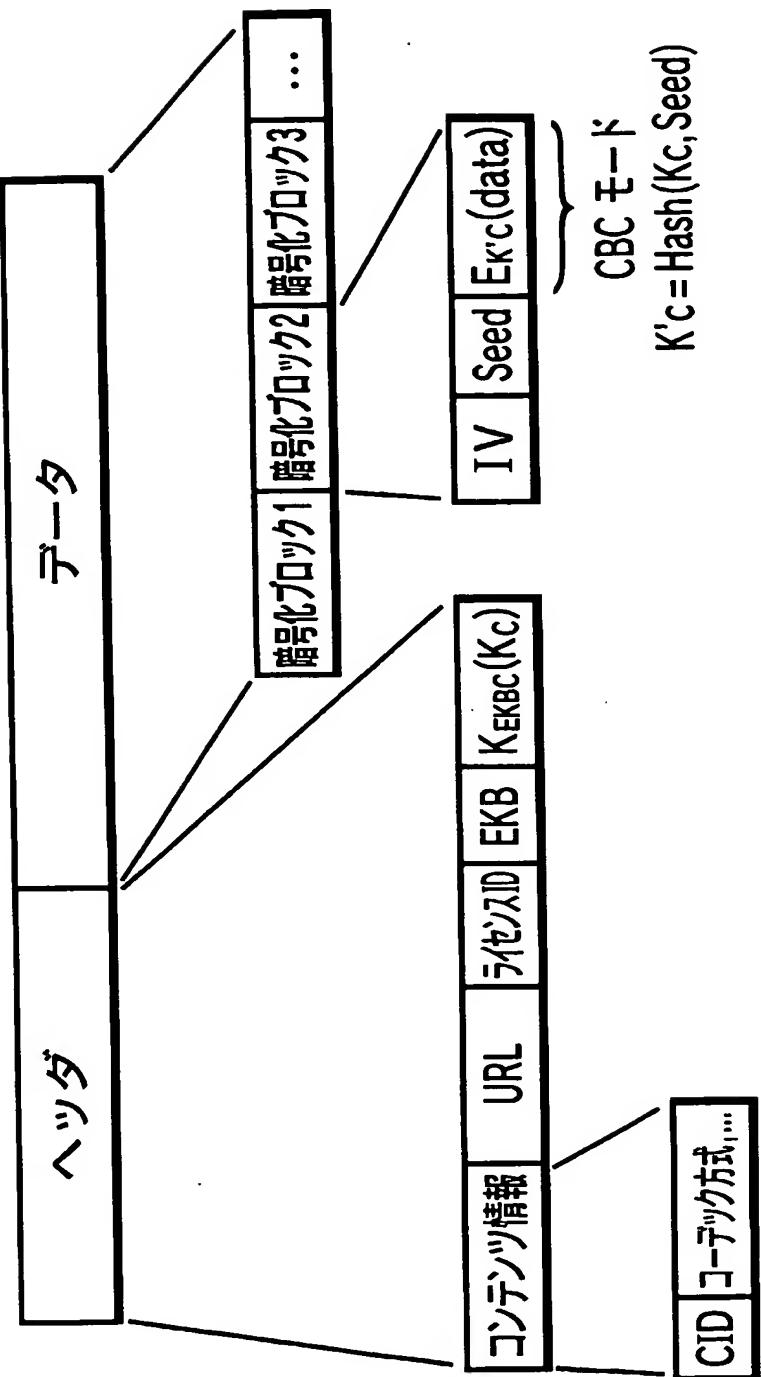


図 6

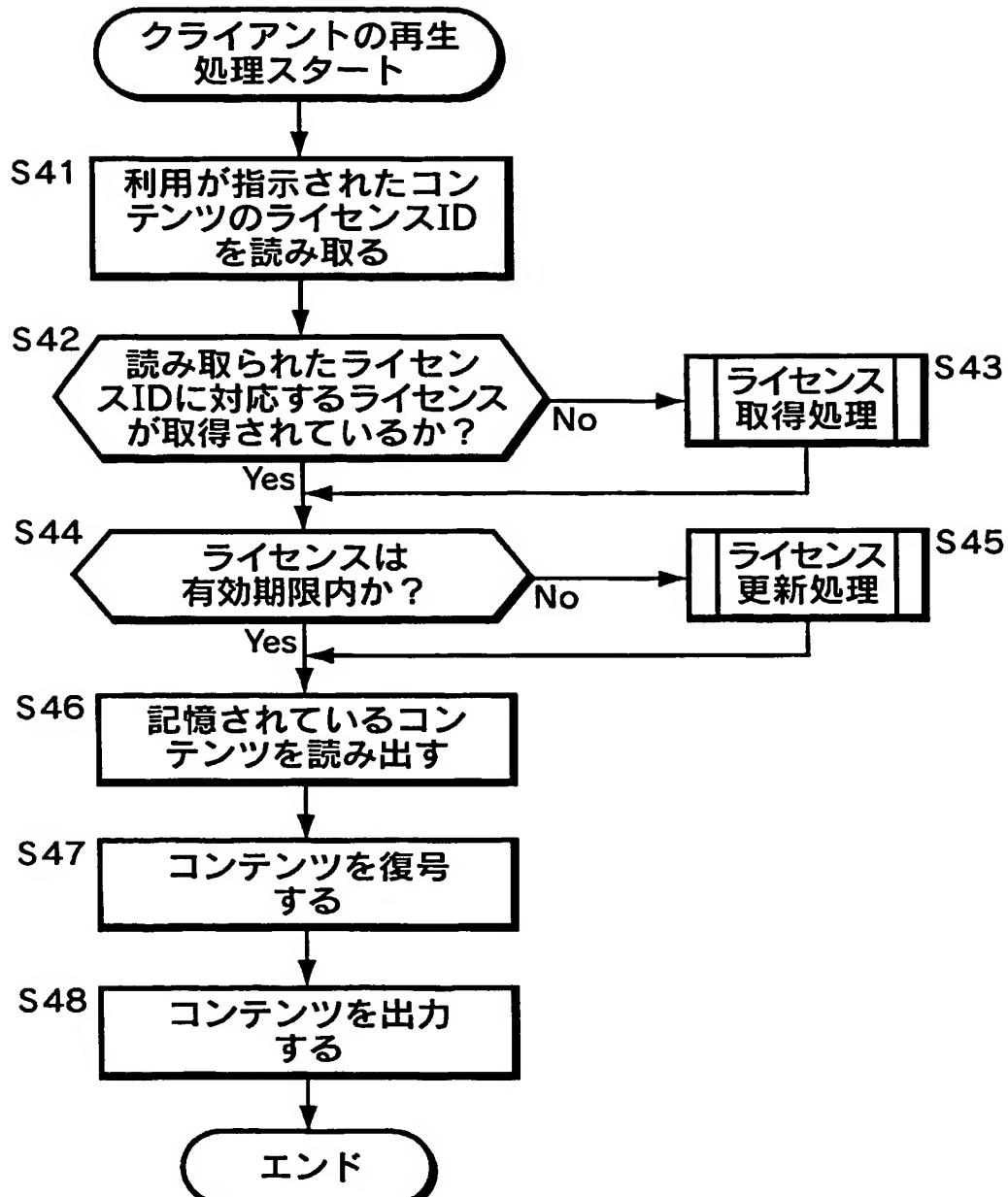
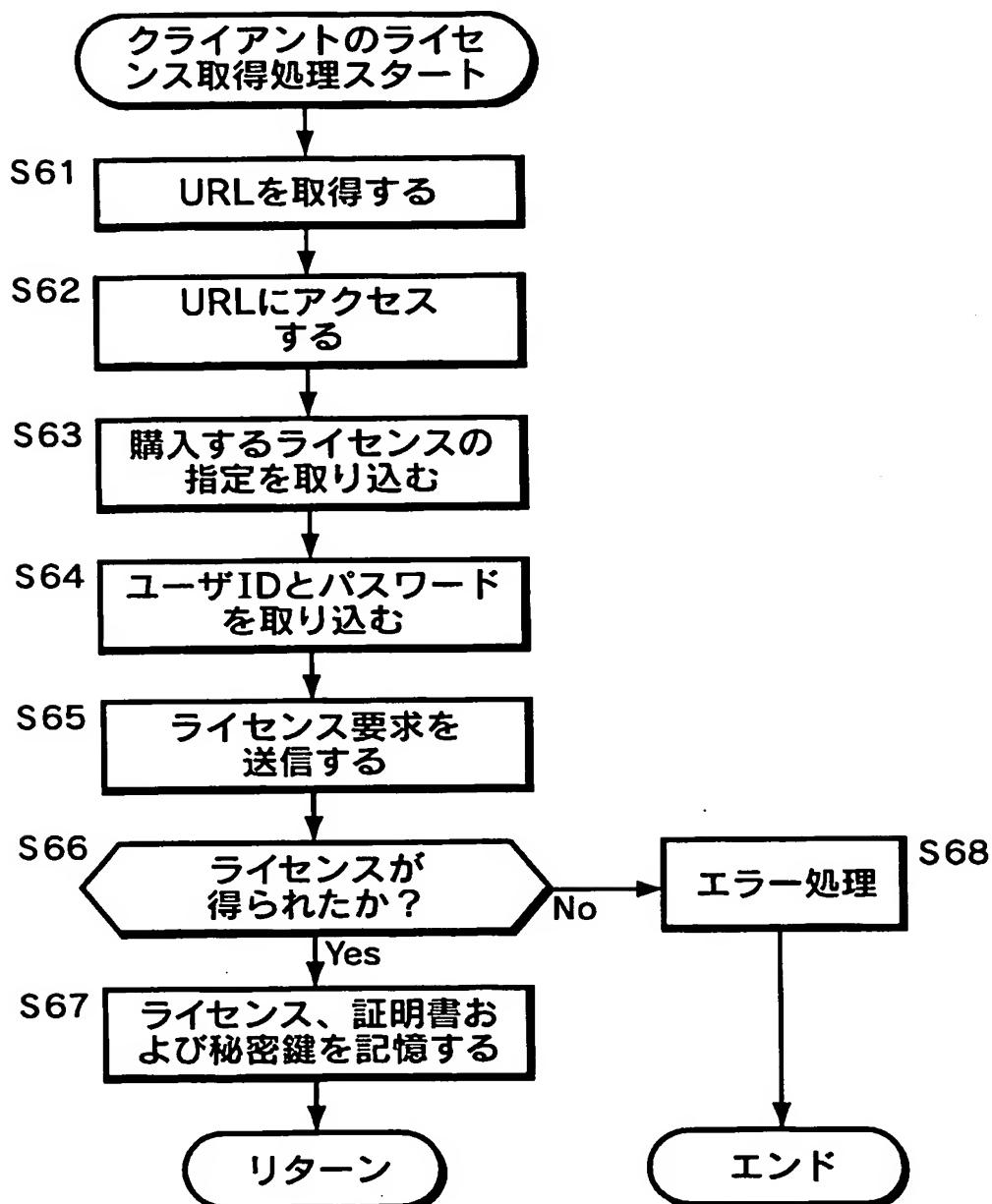
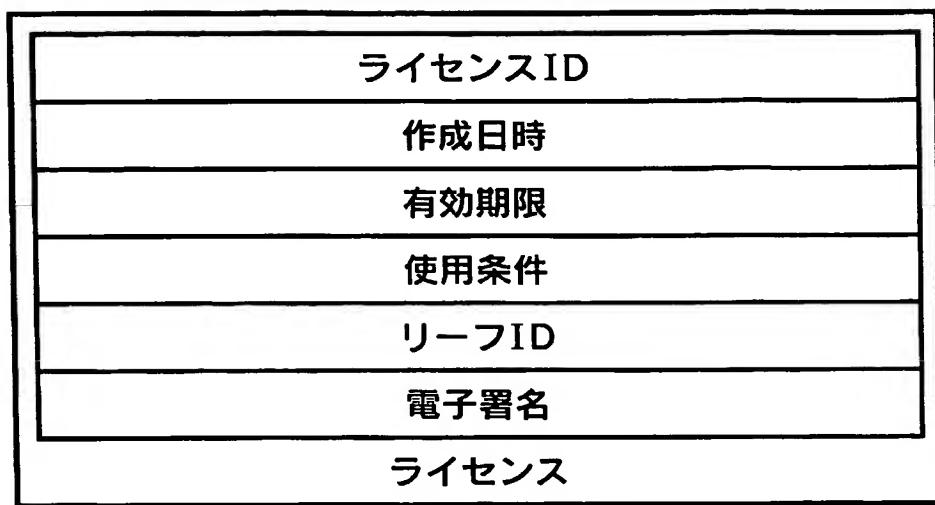


図 7



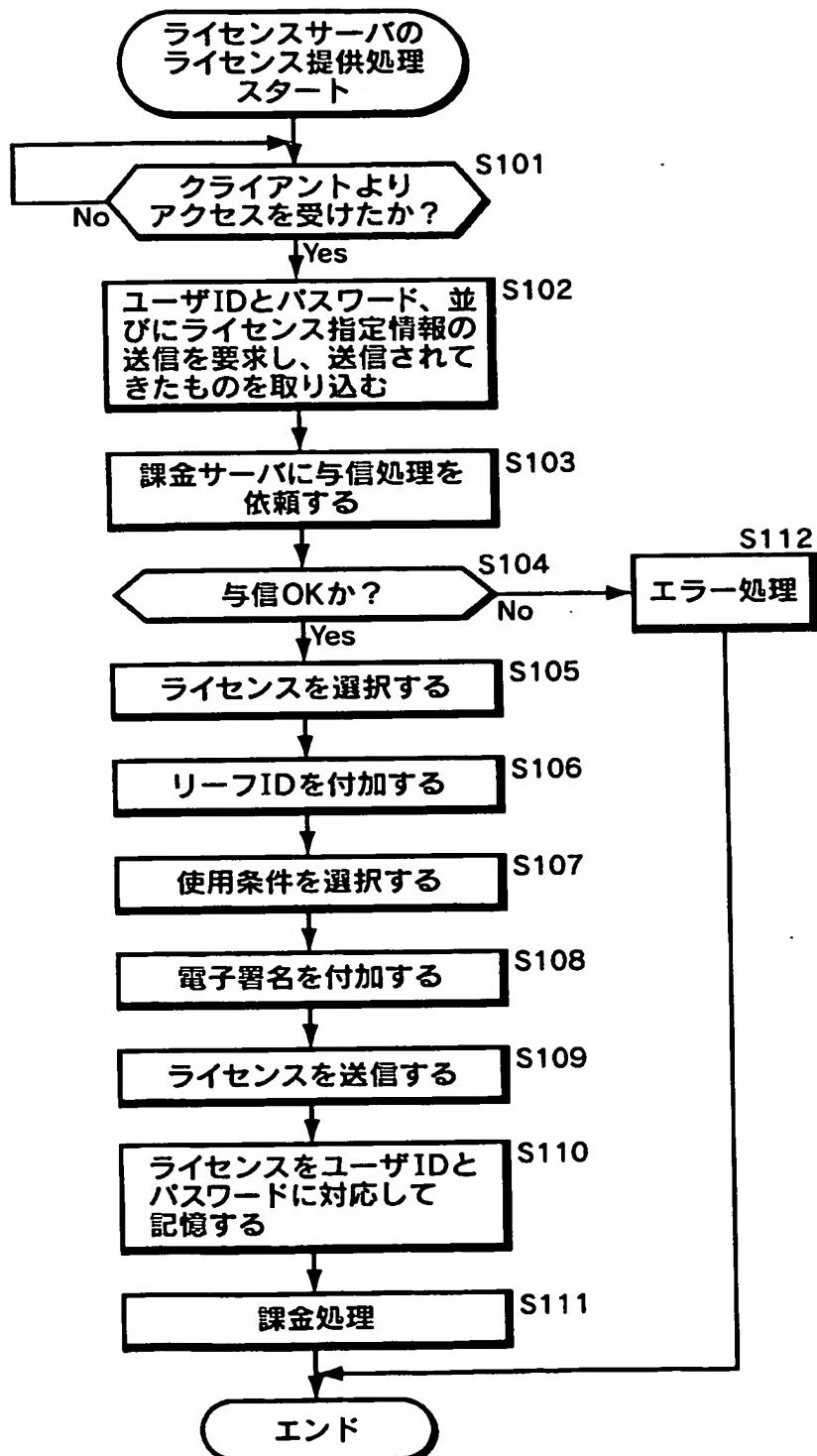
8/45

図 8



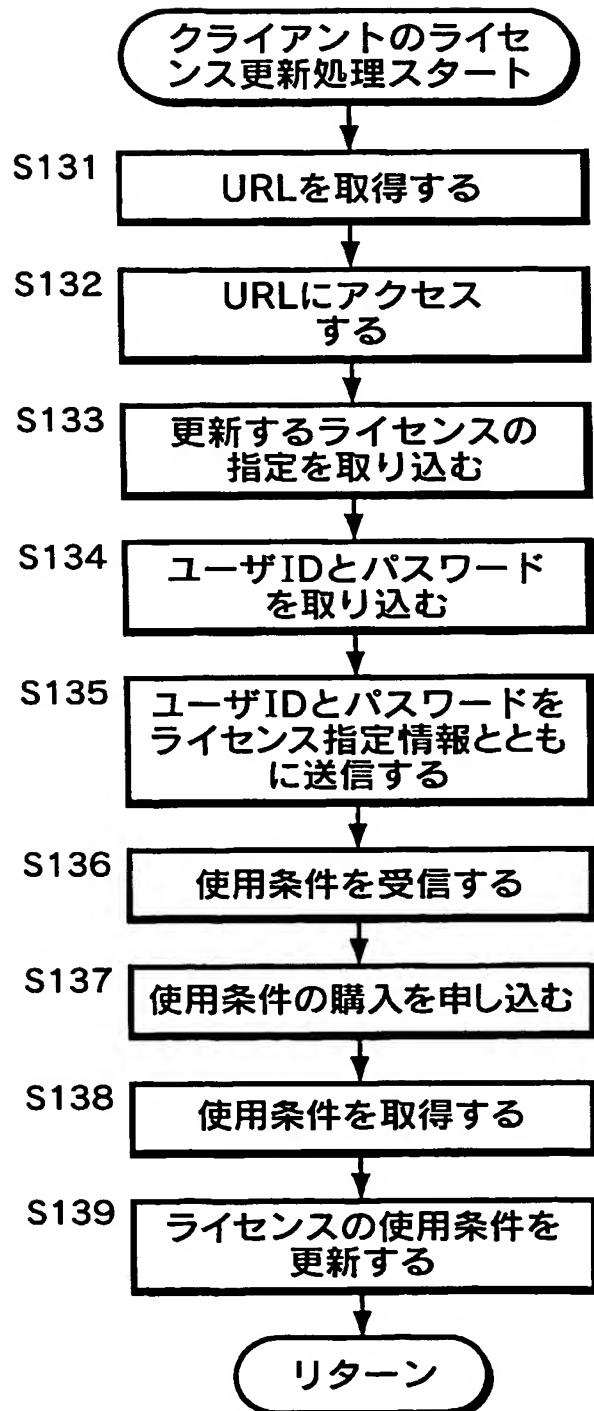
9/45

図9



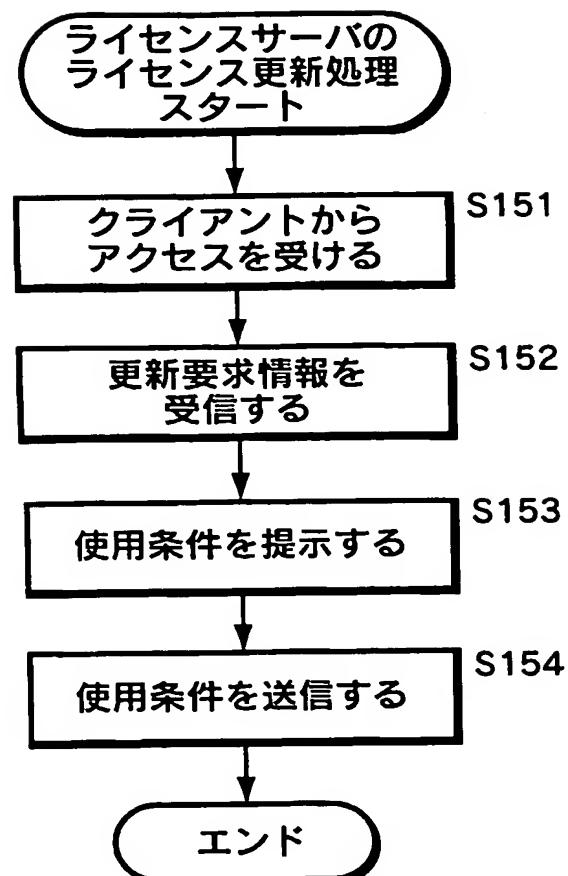
10/45

図 10



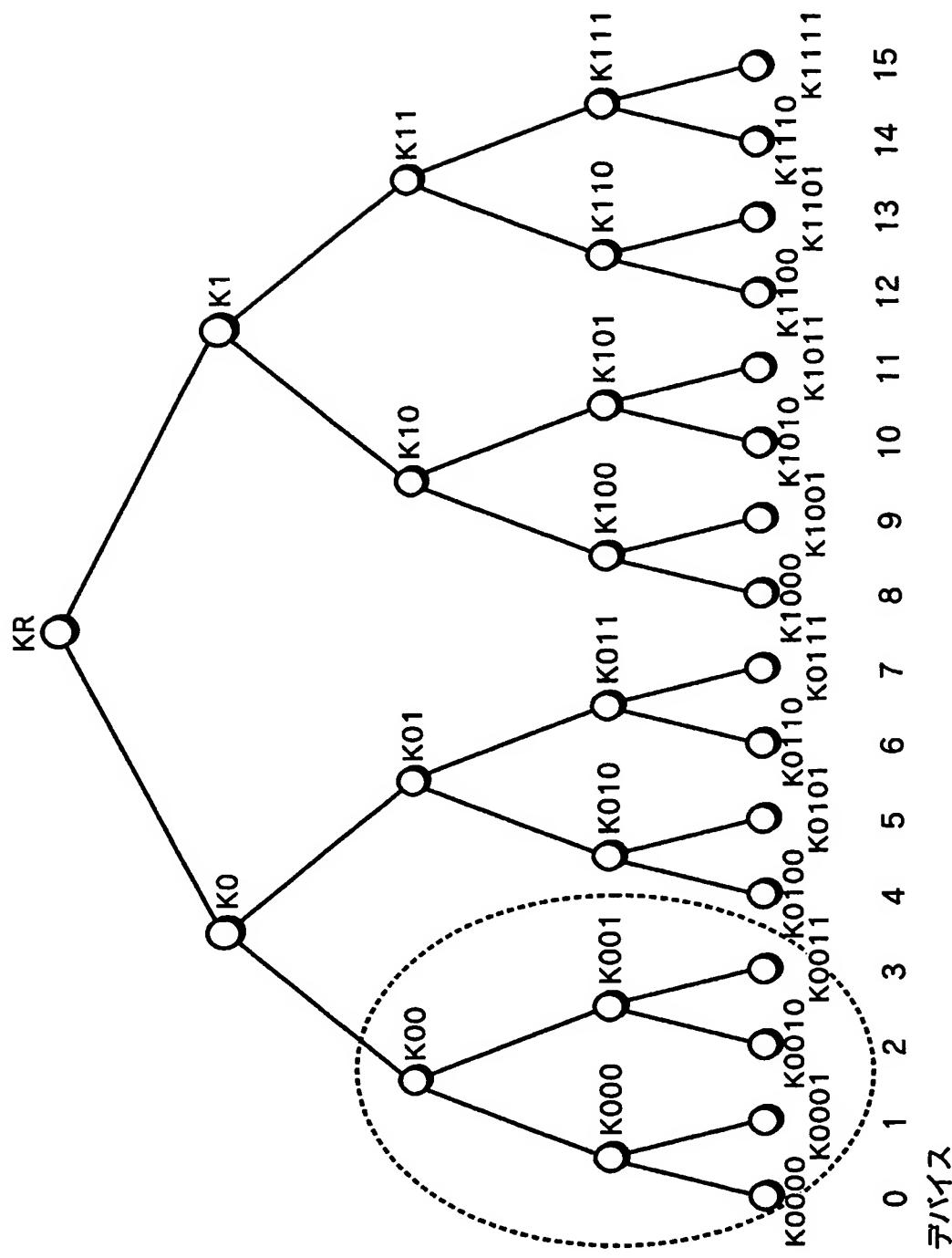
11/45

図 11



12/45

図12



13/45

図13

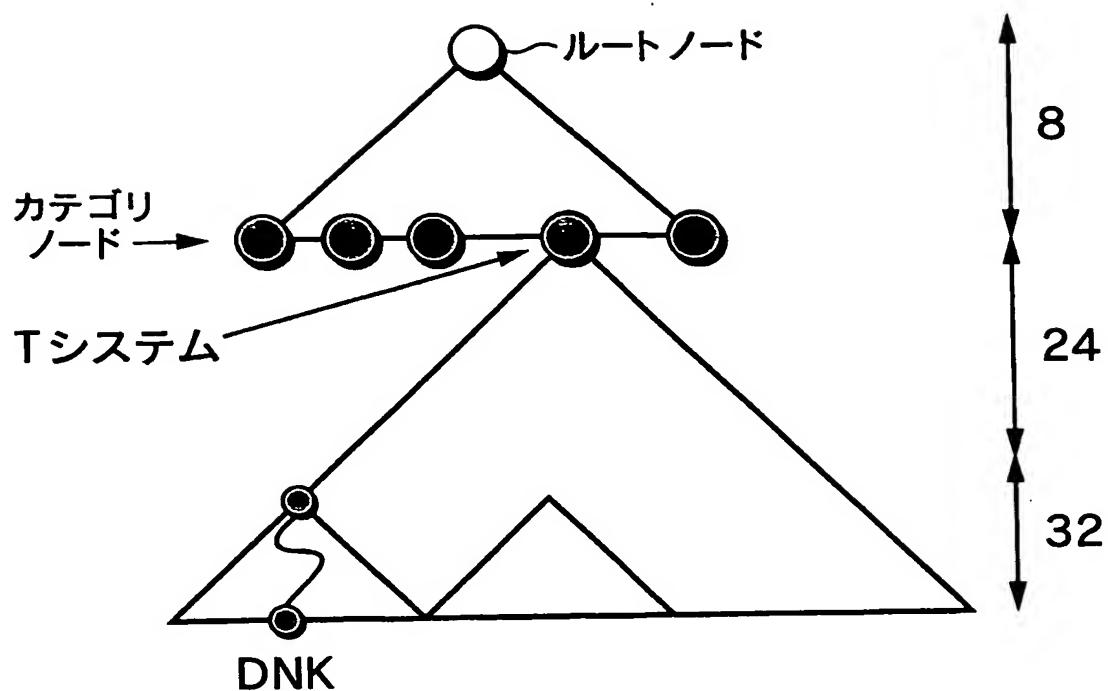
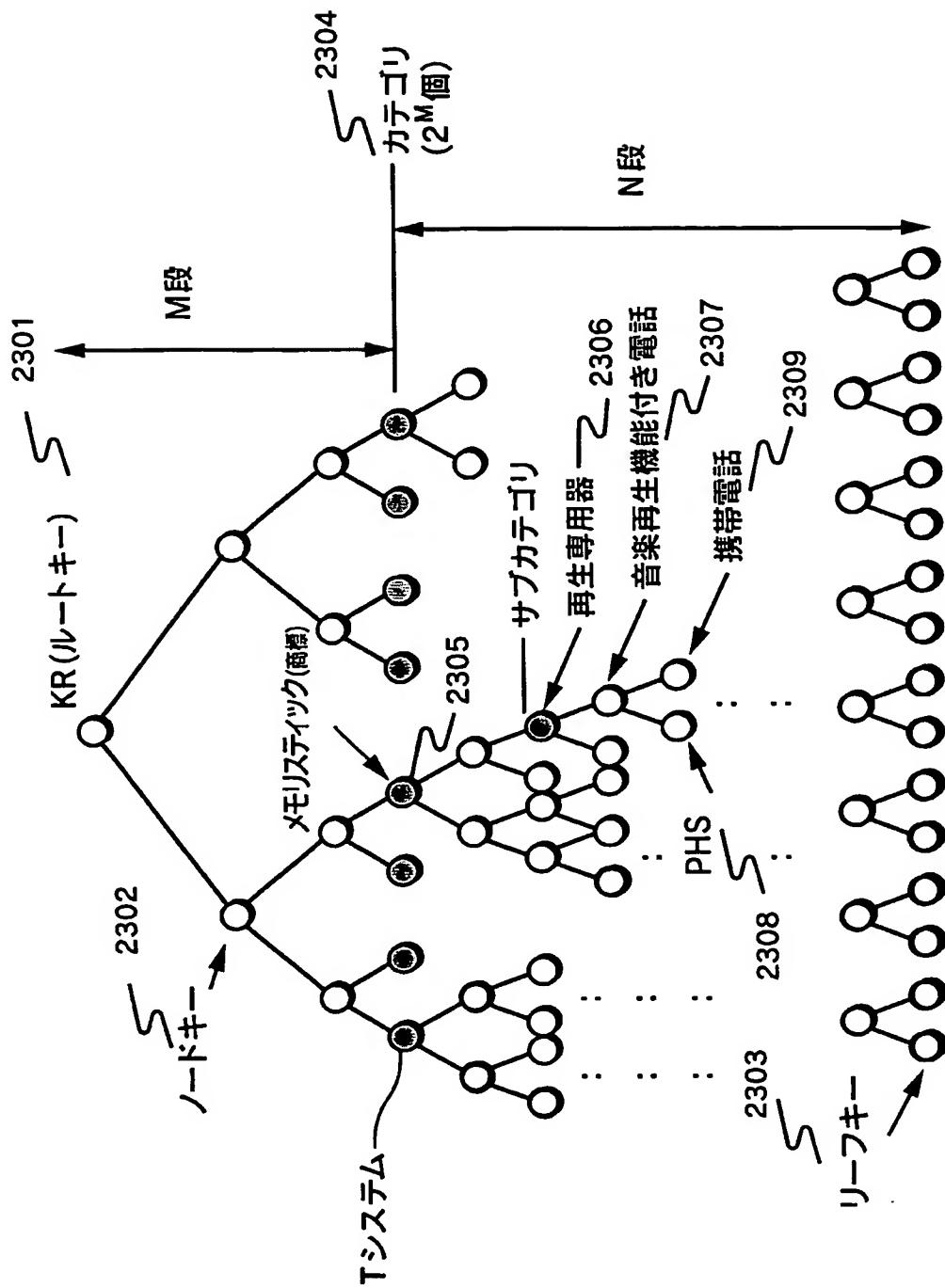


図 14



15/45

図15A

バージョン(Version) t	
インデックス	暗号化キー
0	Enc(K(t)0, K(t)R)
00	Enc(K(t)00, K(t)0)
000	Enc(K000, K(t)00)
001	Enc(K(t)001, K(t)00)
0010	Enc(K0010, K(t)001)

図15B

バージョン(Version) t	
インデックス	暗号化キー
000	Enc(K000, K(t)00)
001	Enc(K(t)001, K(t)00)
0010	Enc(K0010, K(t)001)

図16

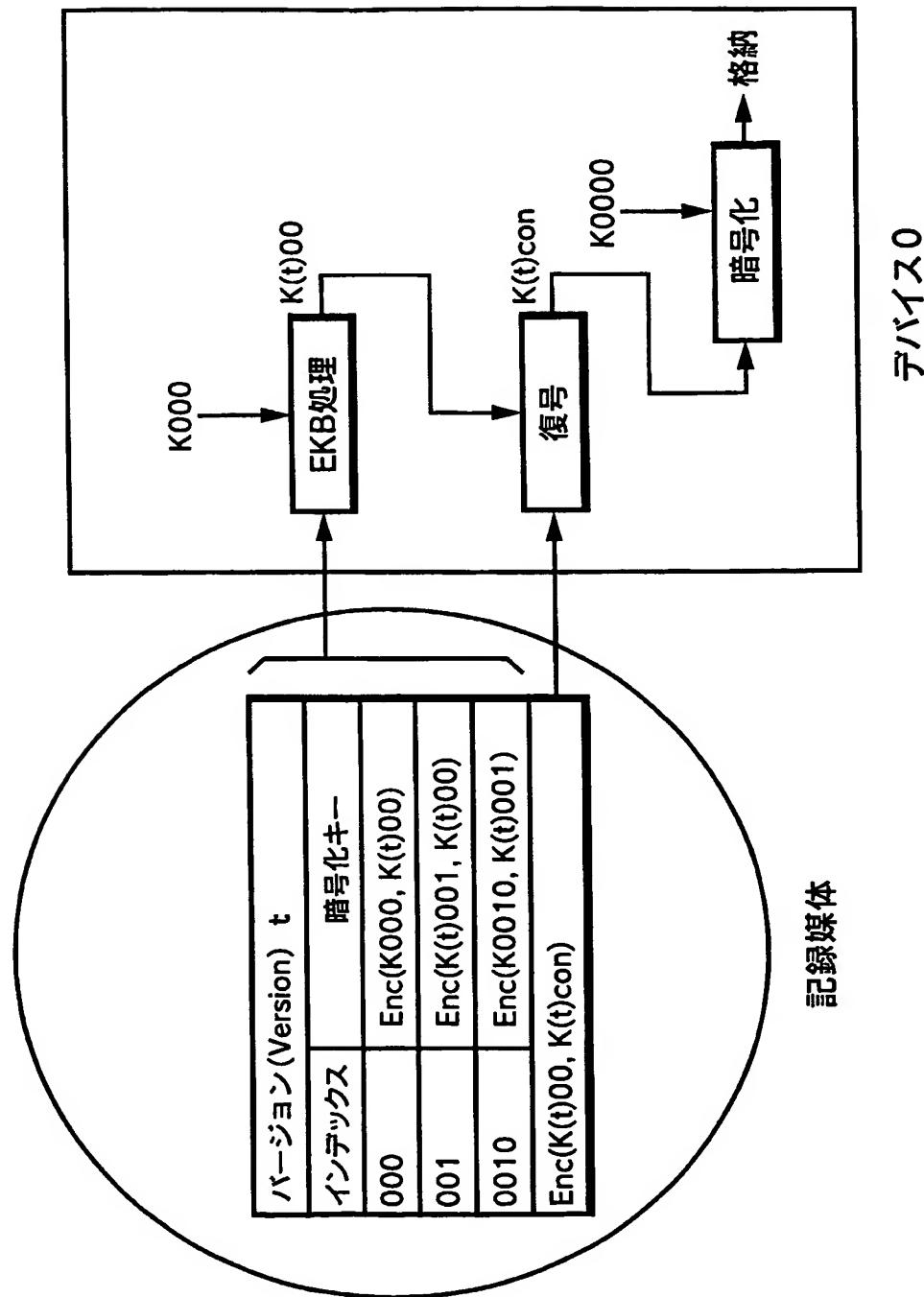


図 17

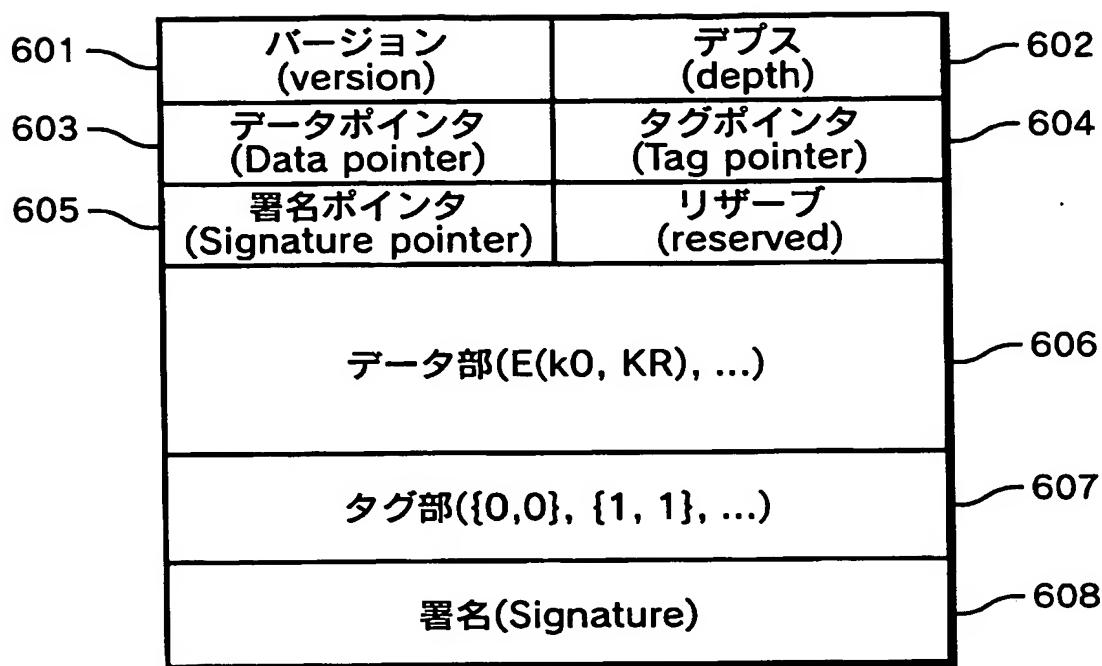
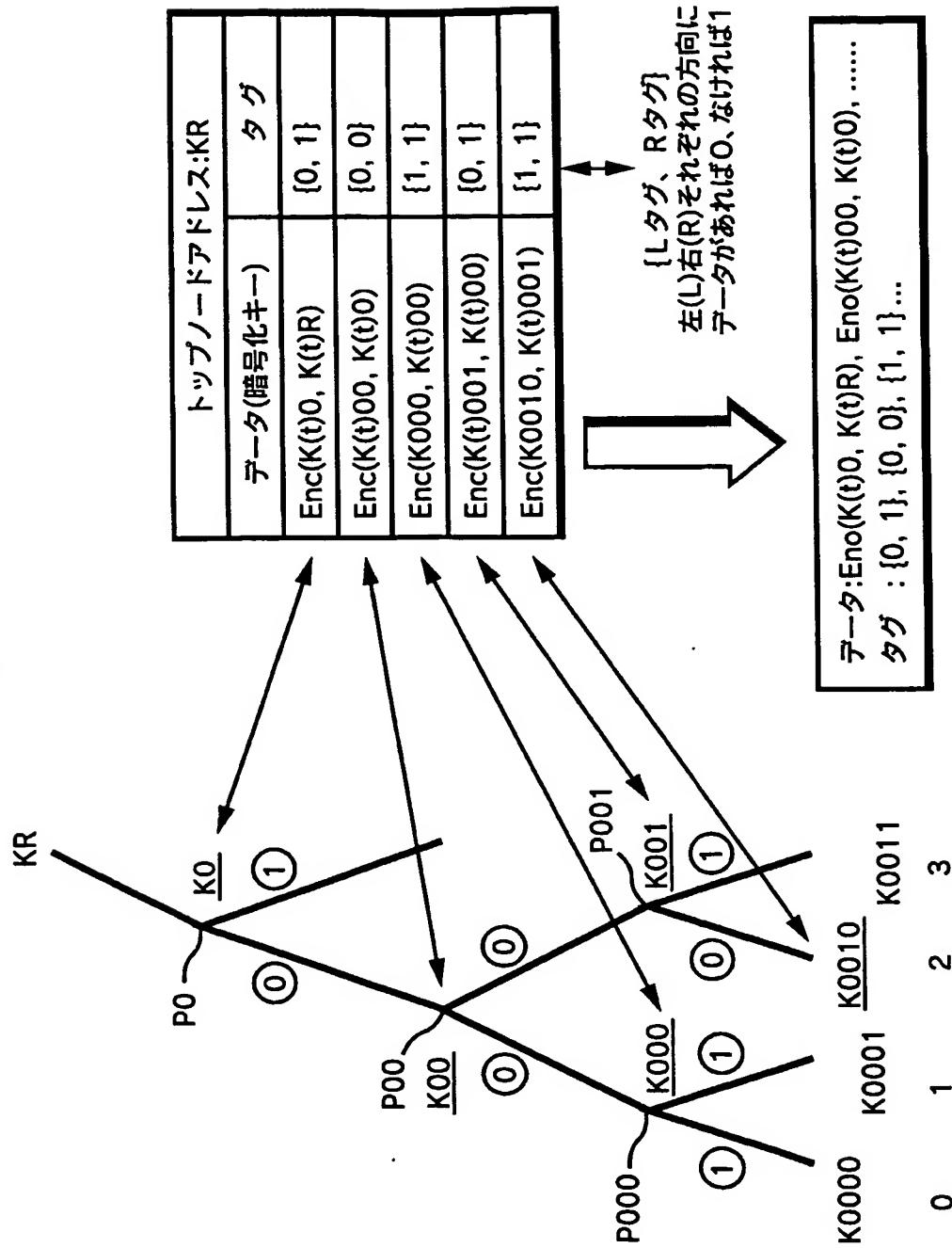


図18



19/45

図 19

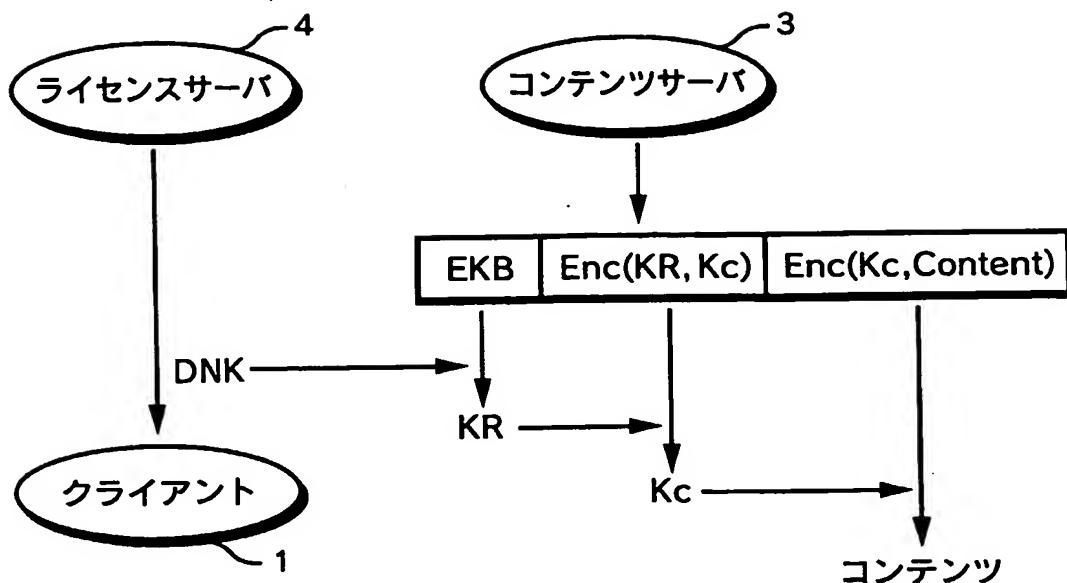


図 20

EKB

 $Enc(DNK, KR)$

20/45

図21

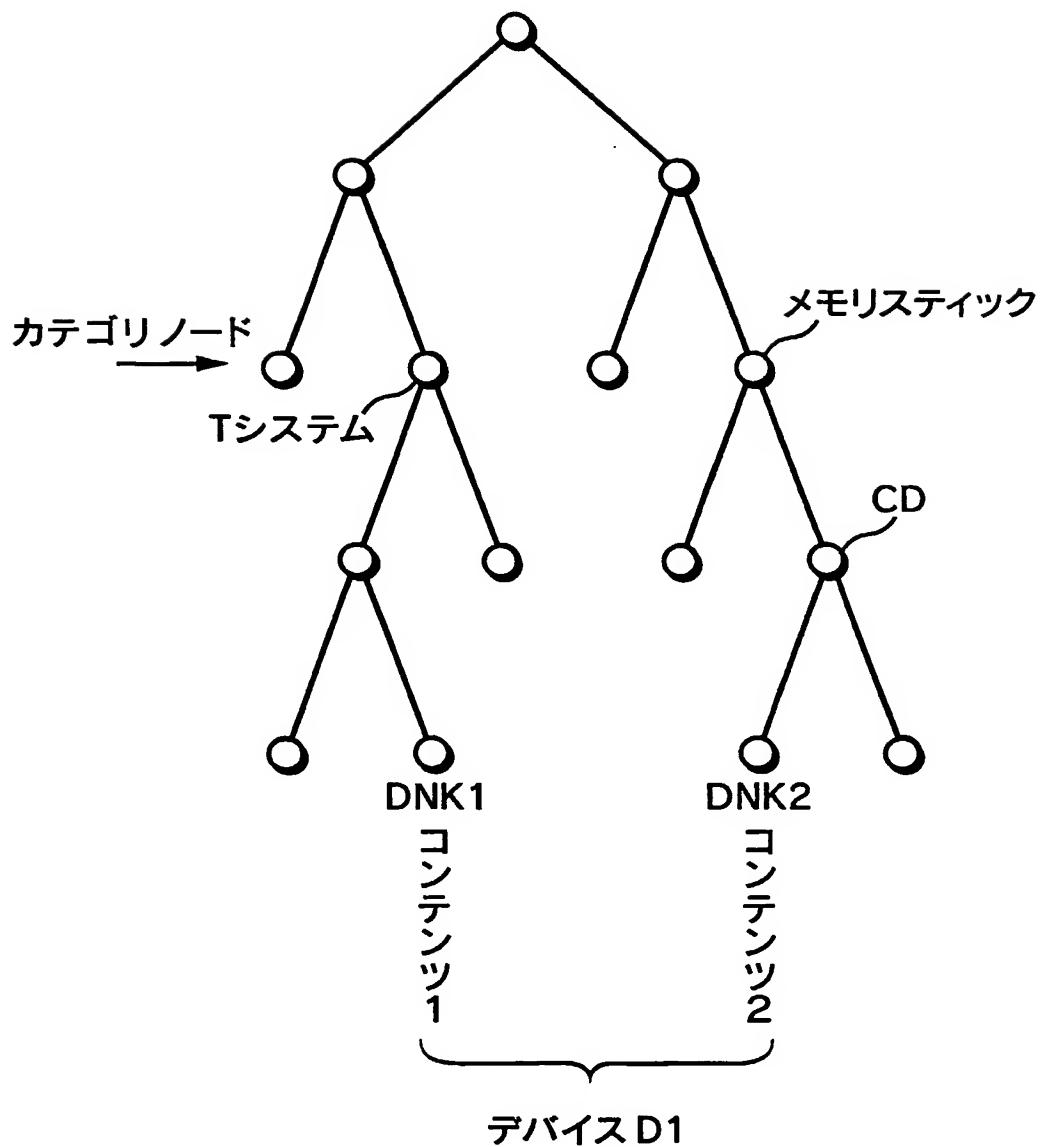
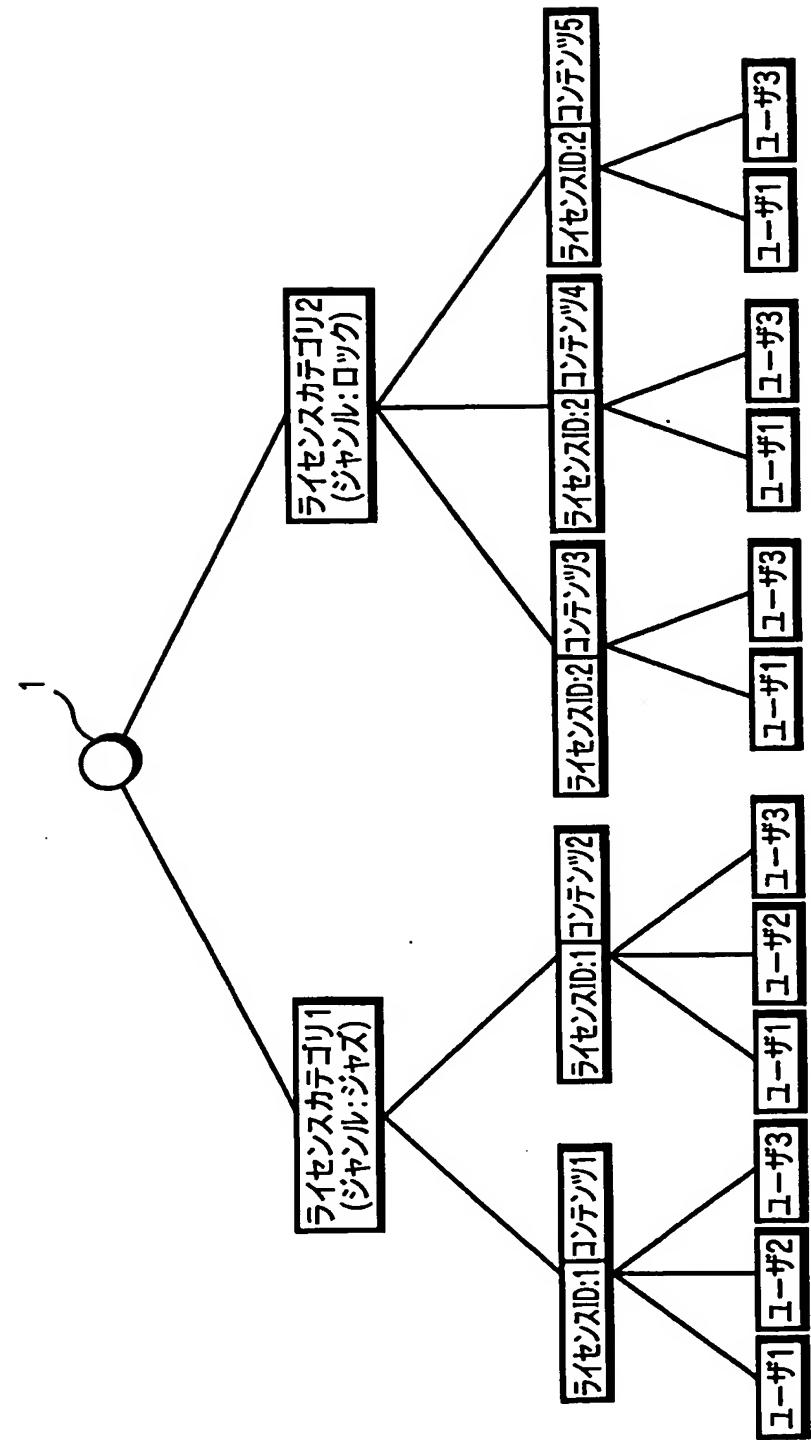


図 22



23

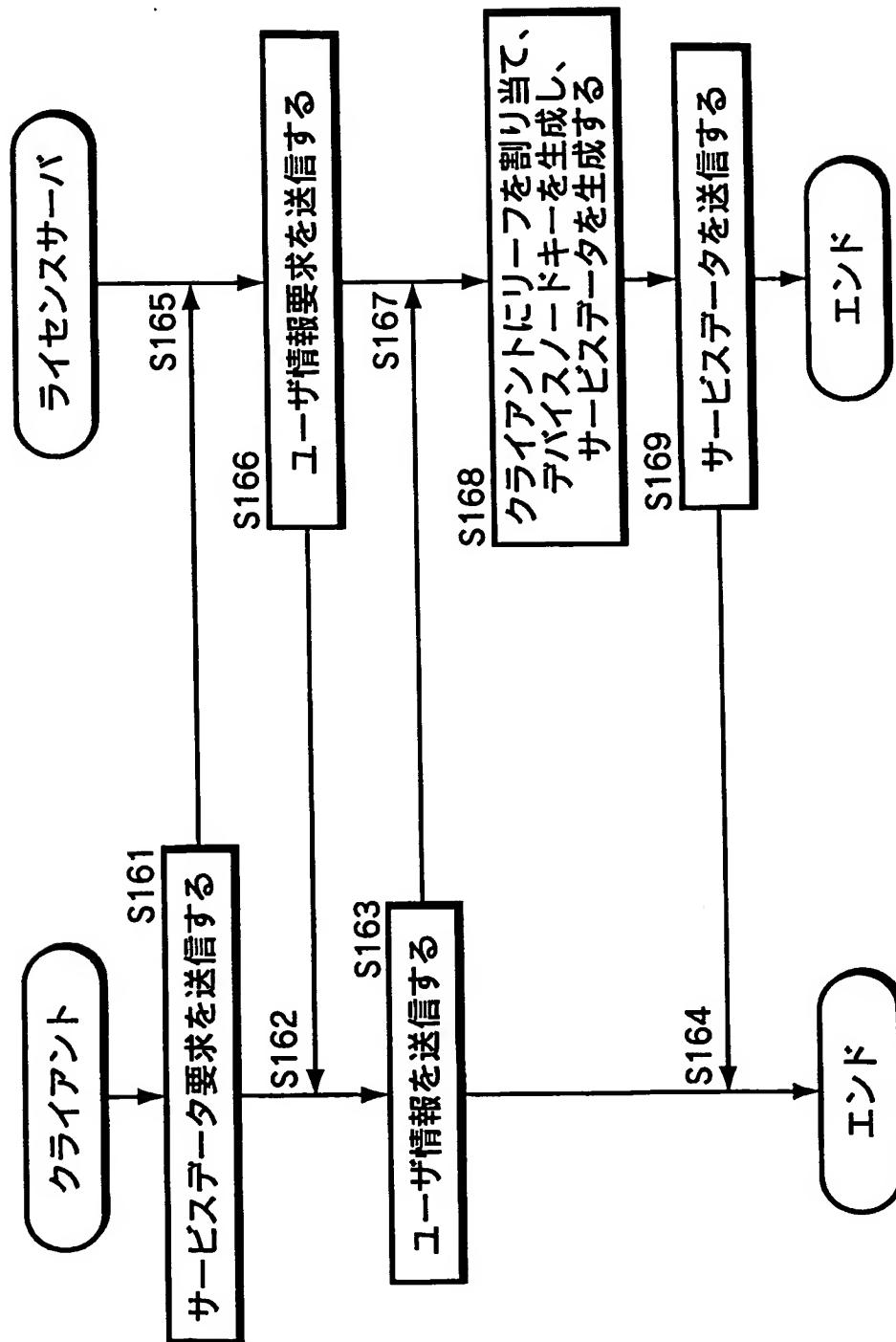
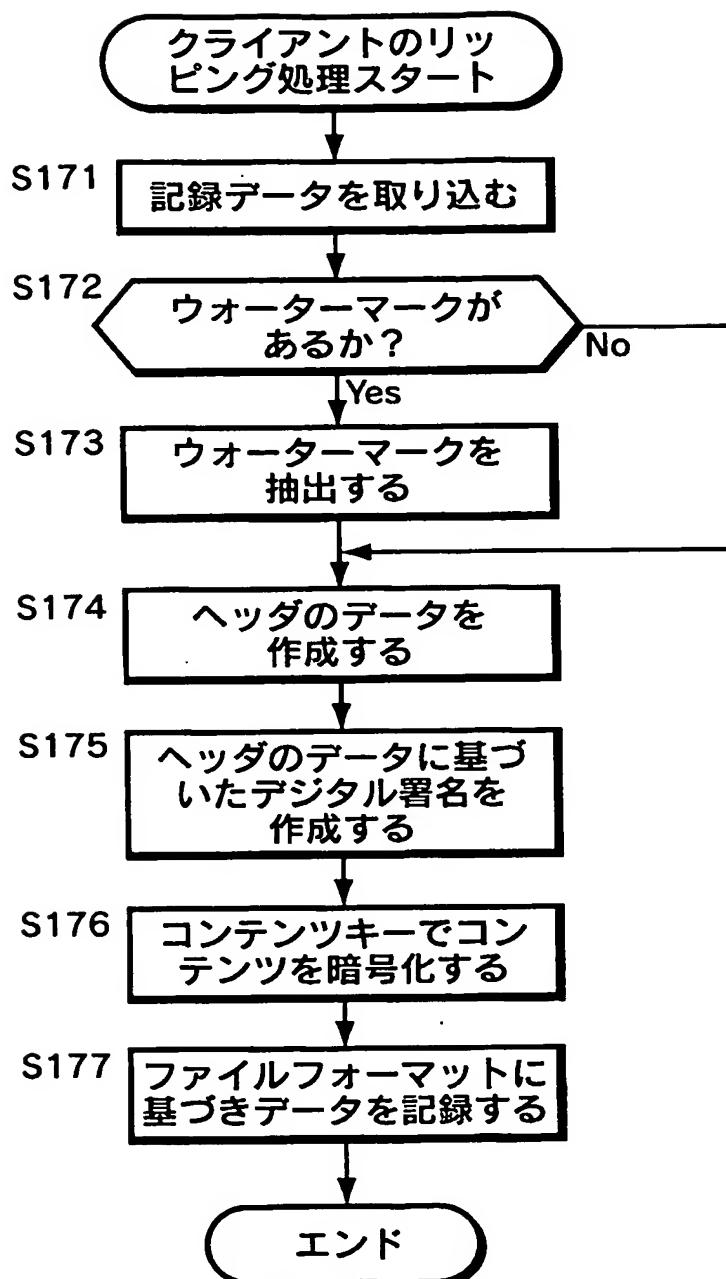


図 24



24/45

図 25

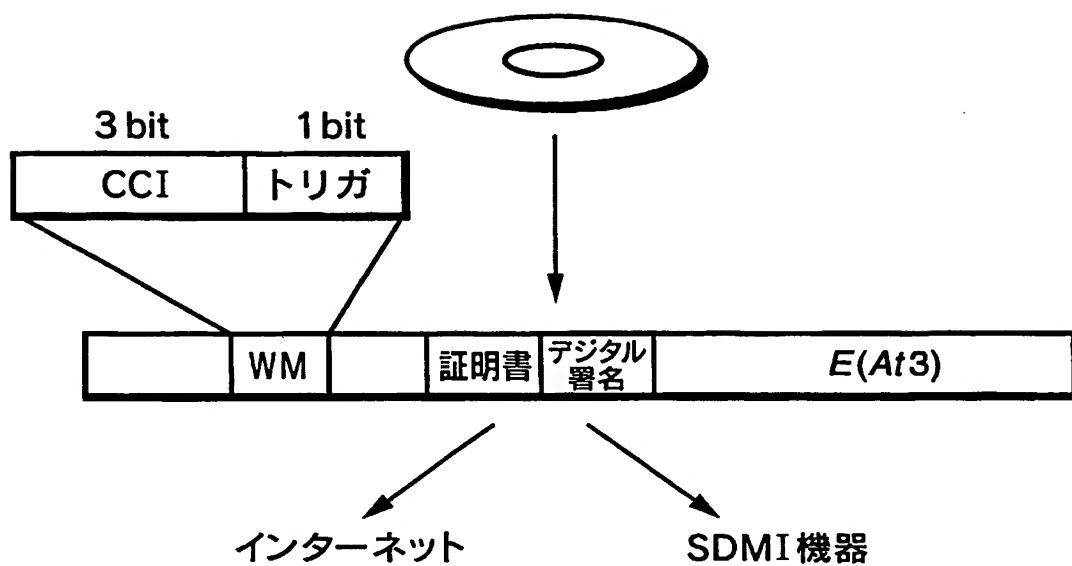


図 26

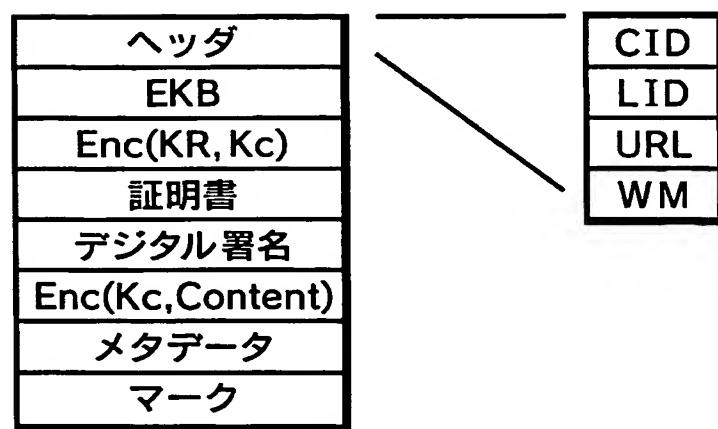
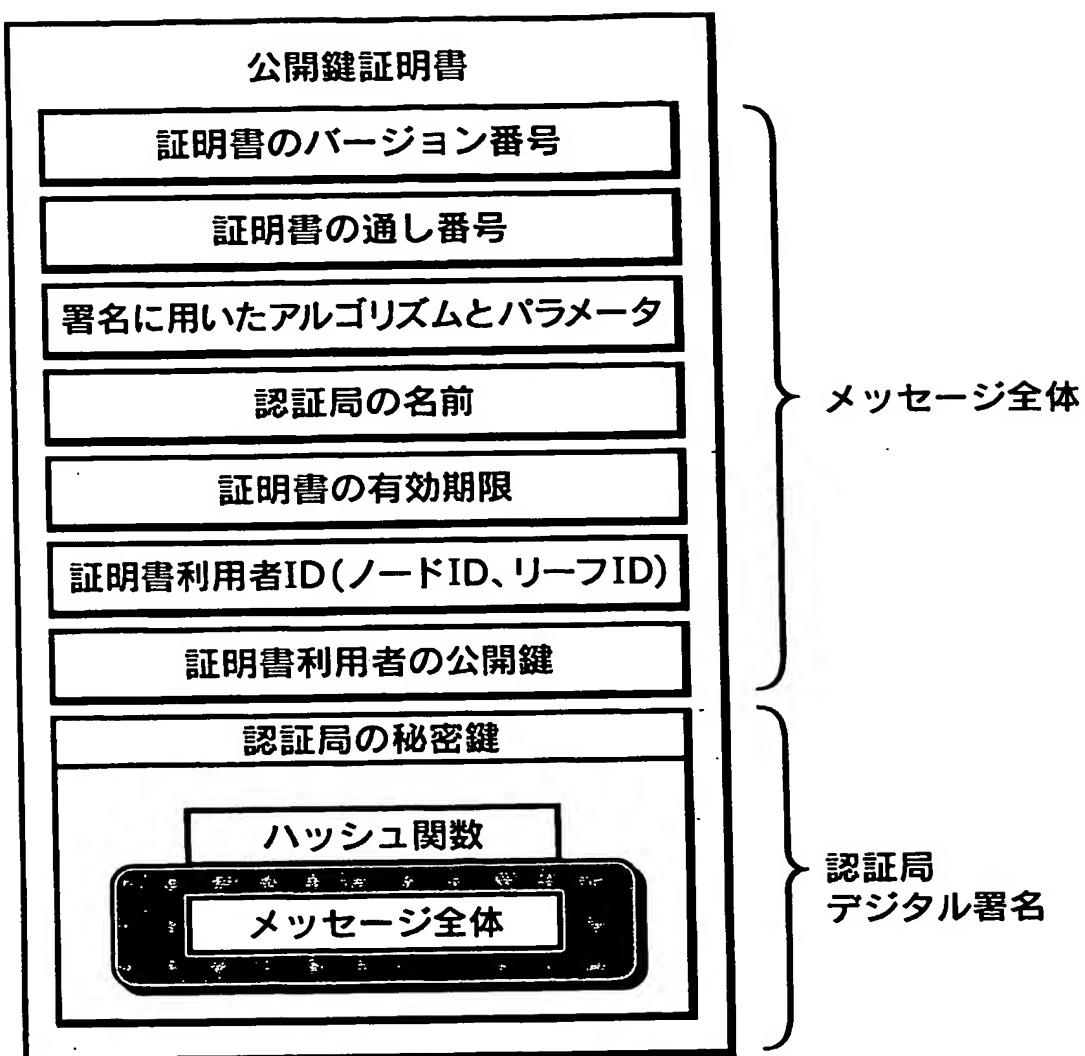


図 27



26/45

図 28

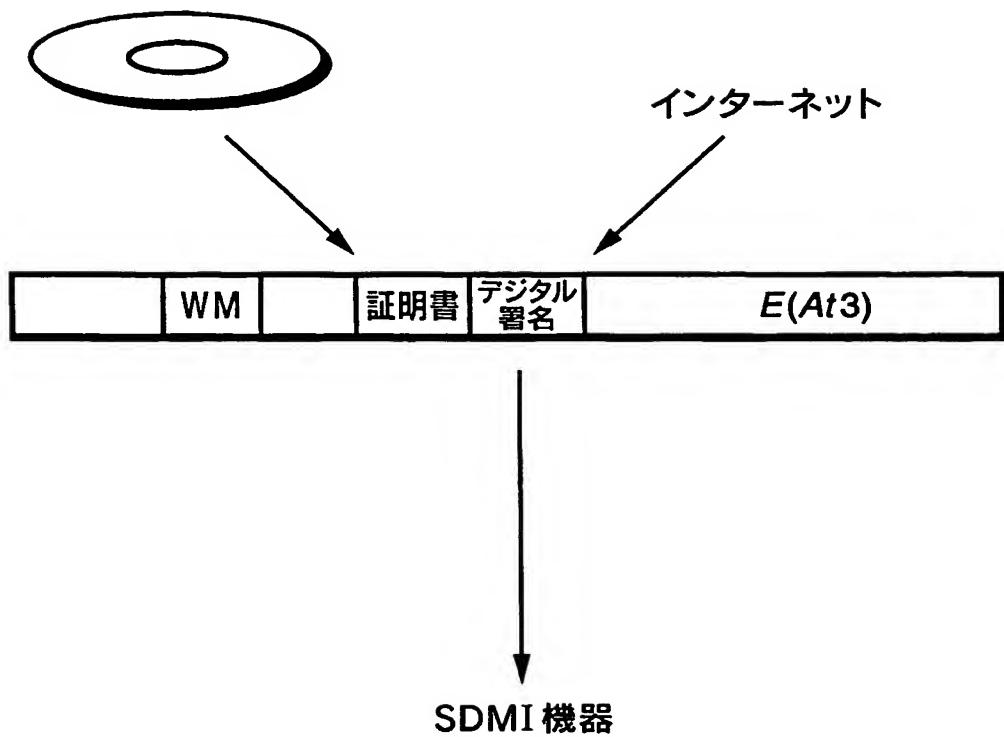


図 29

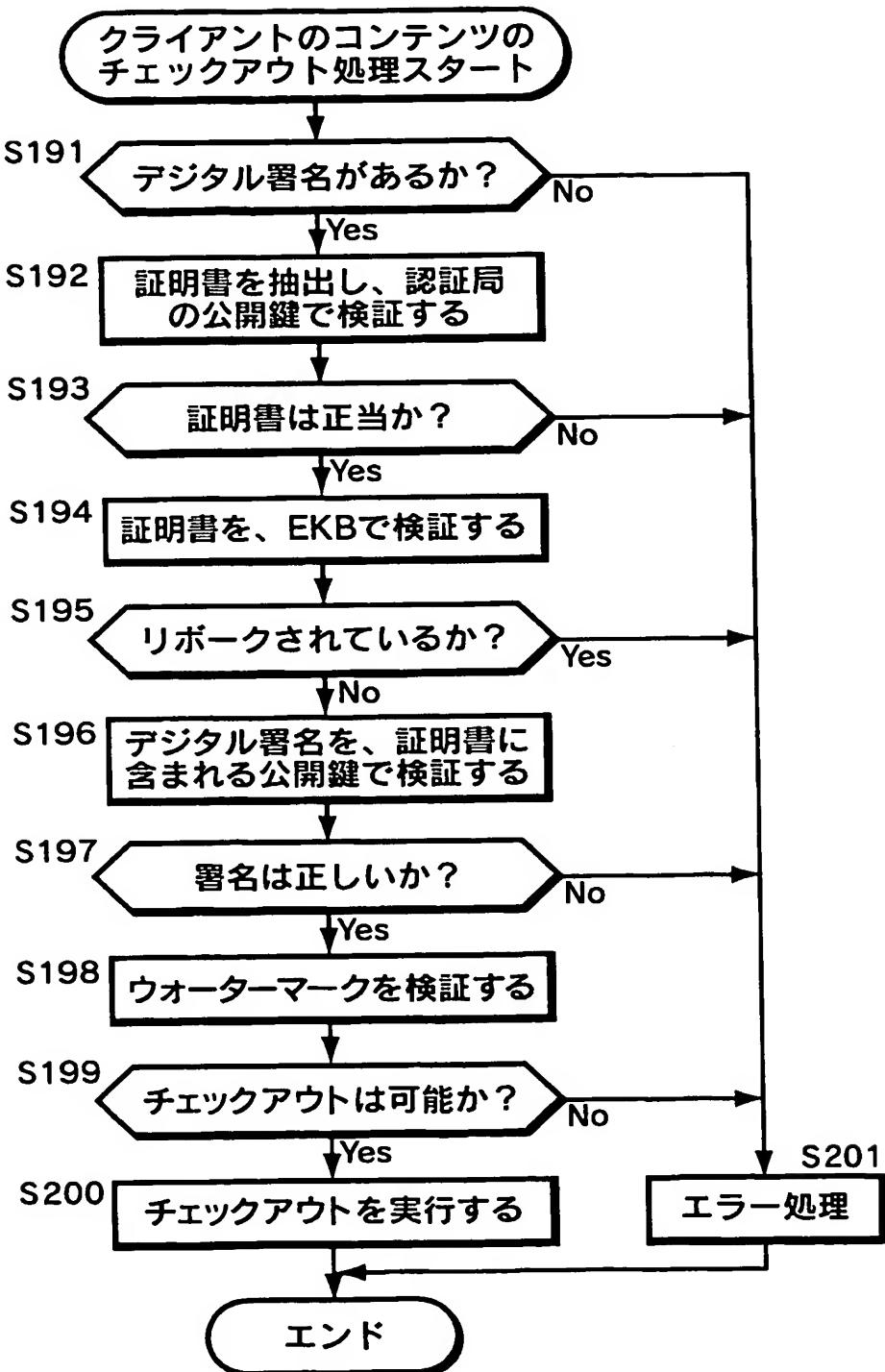
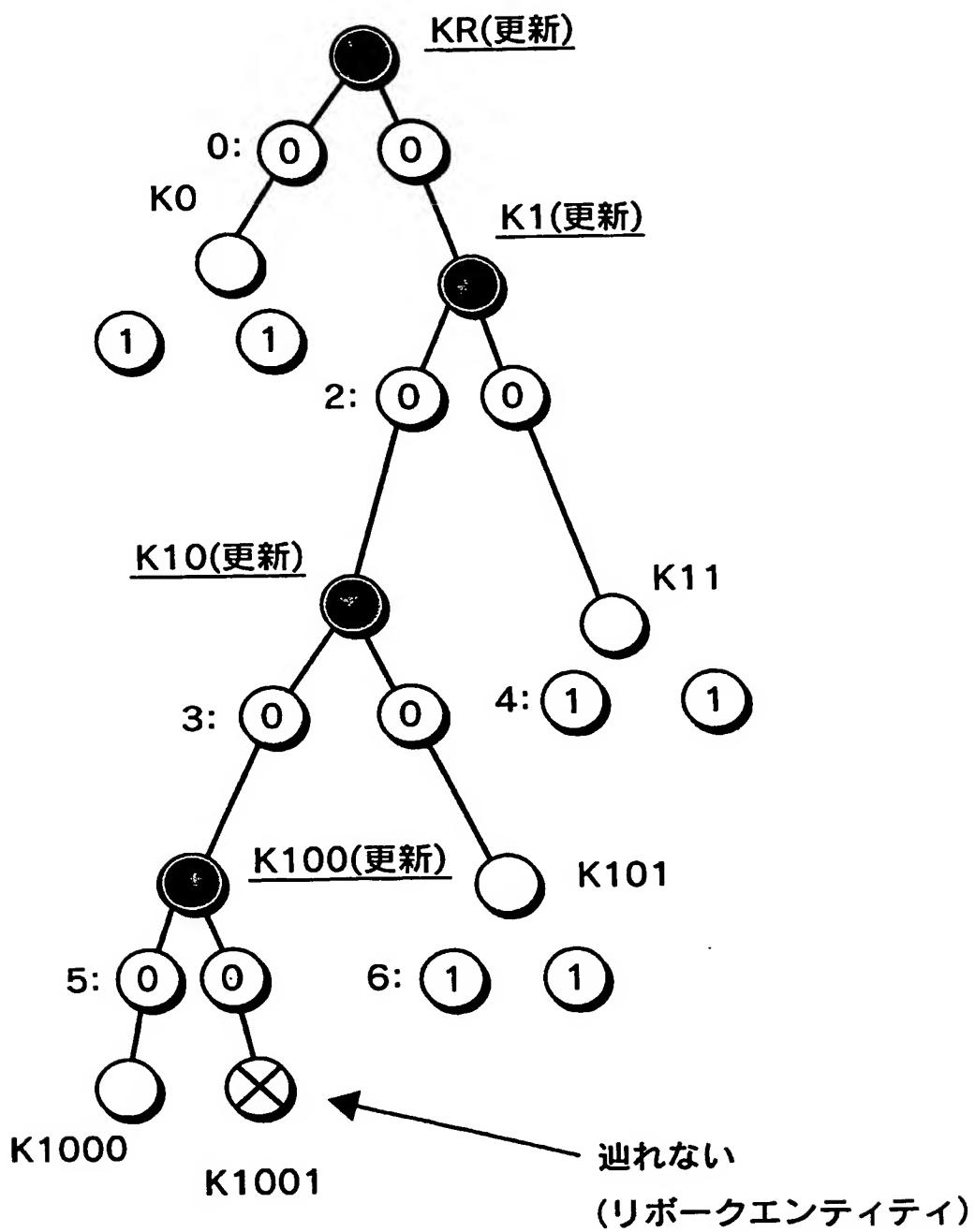


図 30



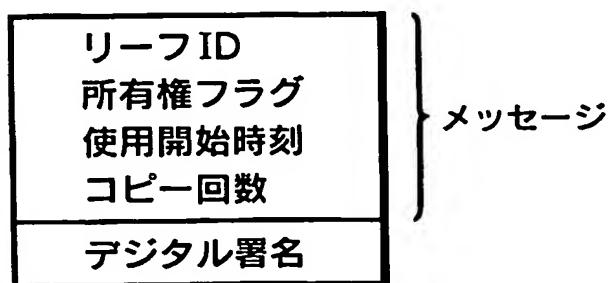
29/45

図 31

データ (暗号化キー)	Enc(K0, K(t)R), Enc(K(t)1, K(t)R) Enc(K(t)10, K(t)1), Enc(K11, K(t)1) Enc(K(t)100, K(t)10), Enc(K101, K(t)10) Enc(K1000, K(t)100)
タグ	0: {0, 0}, 1:{1, 1}, 2:{0, 0}, 3:(0, 0) 4: {1, 1}, 5:{0, 1}, 6:{1, 1}

↑  
 {L タグ、R タグ}  
 左(L)右(R)それぞれの方向に  
 データがあれば0、なければ1

図 32



30/45

図 33

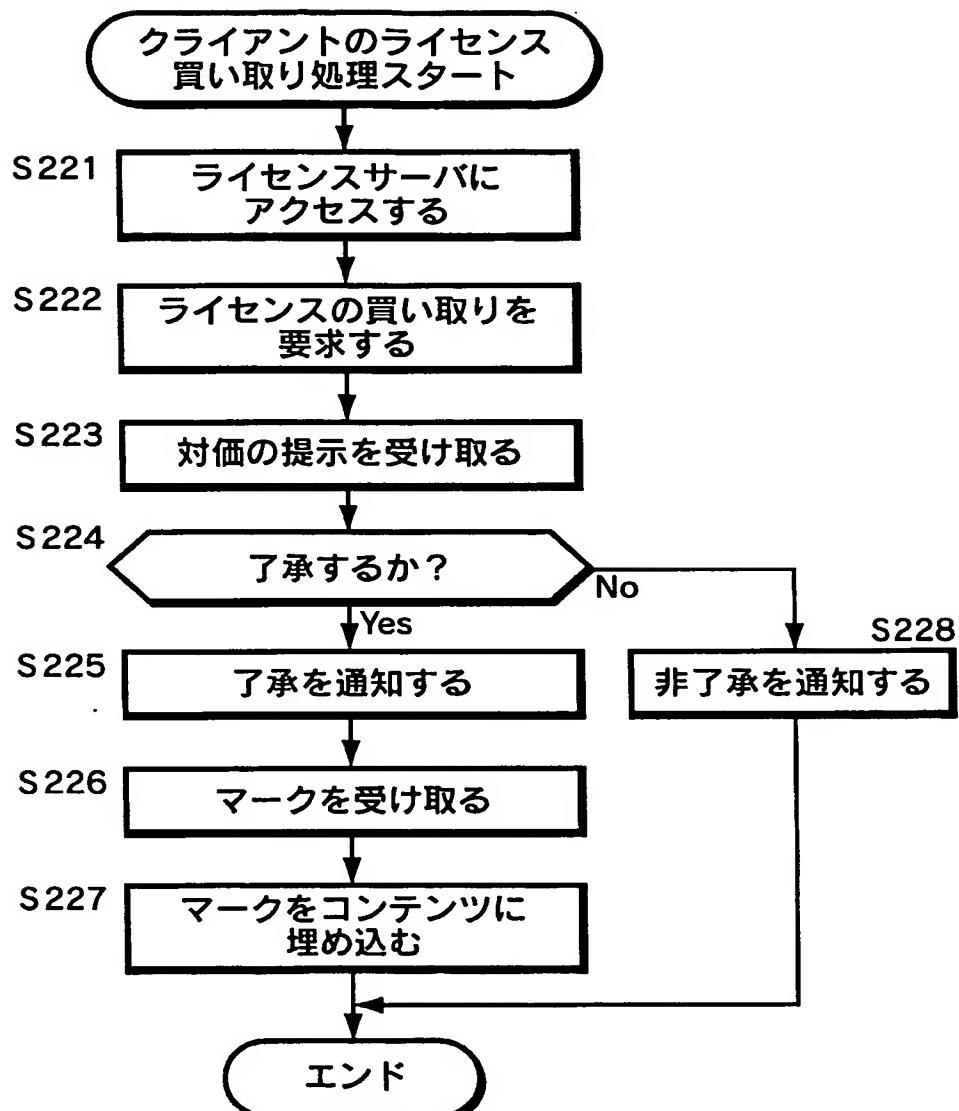
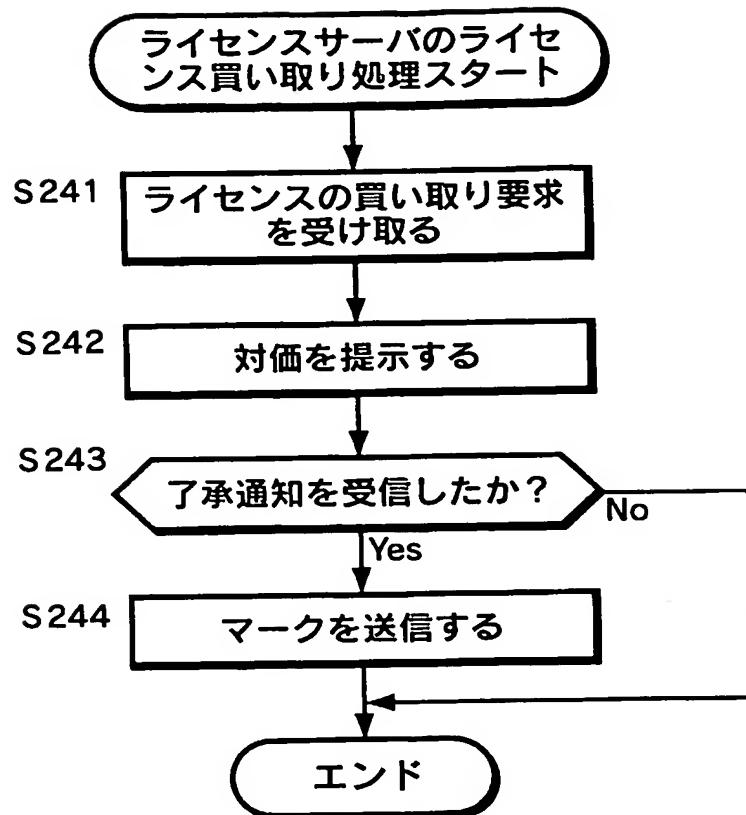


図 34

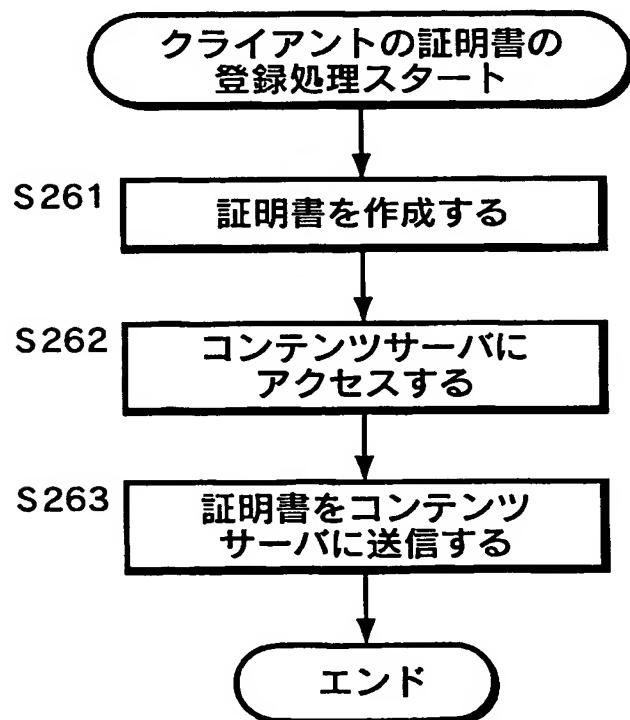


32/45

図 35

Mark = {LeafID, Own, Sigs(LeafID, Own)}

図 36



33/45

図 37

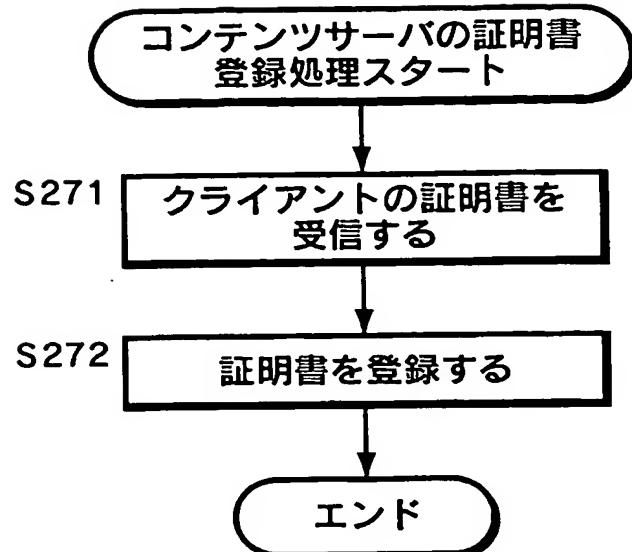
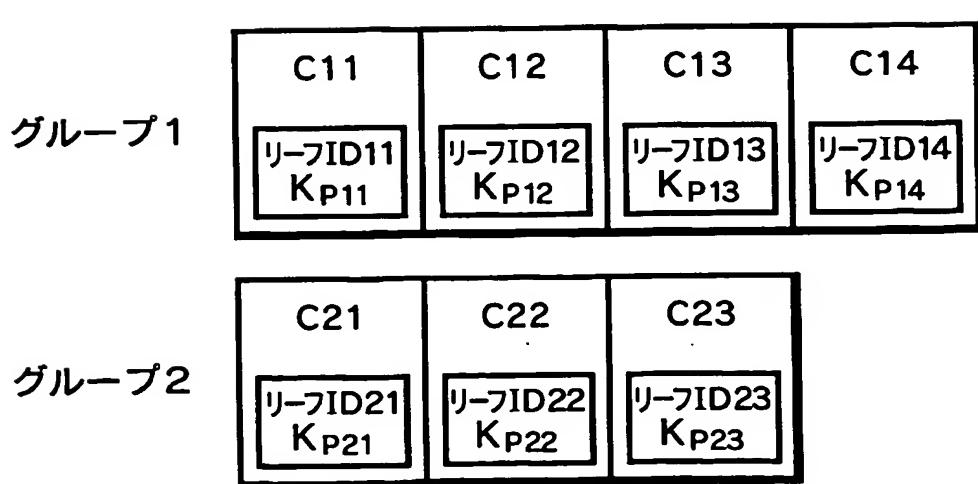


図 38



34/45

図 39

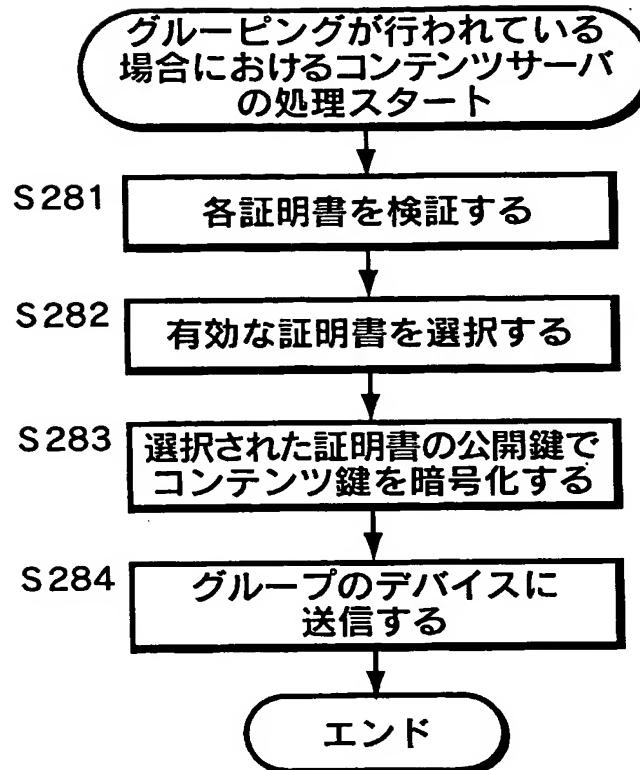


図 40

$$\text{Enc}(K_{P11}, K_C), \text{Enc}(K_{P12}, K_C), \text{Enc}(K_{P13}, K_C)$$

図 41

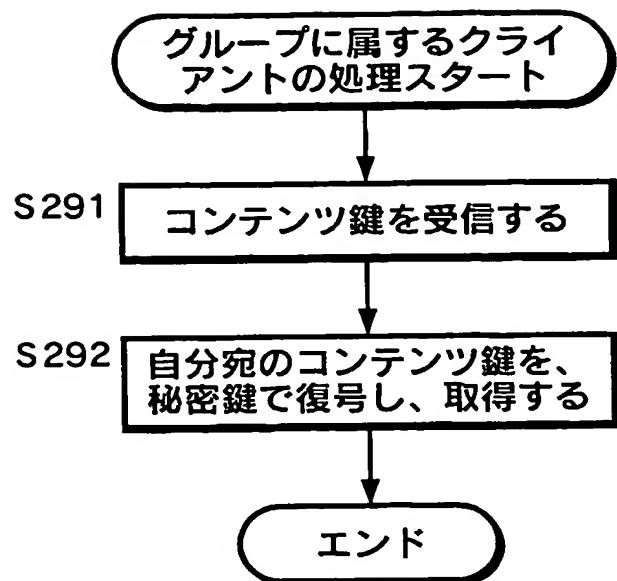


図 42

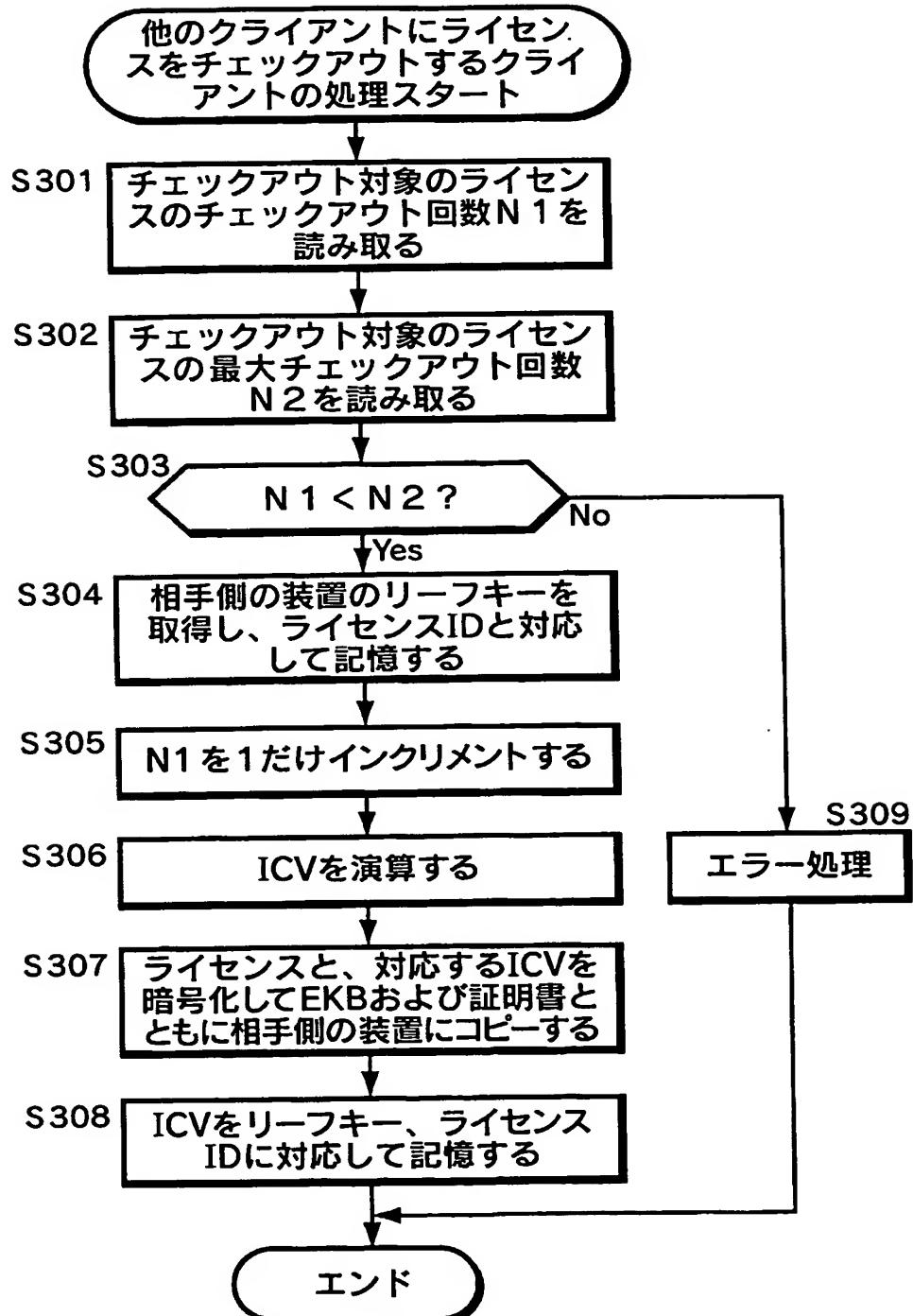


図 43

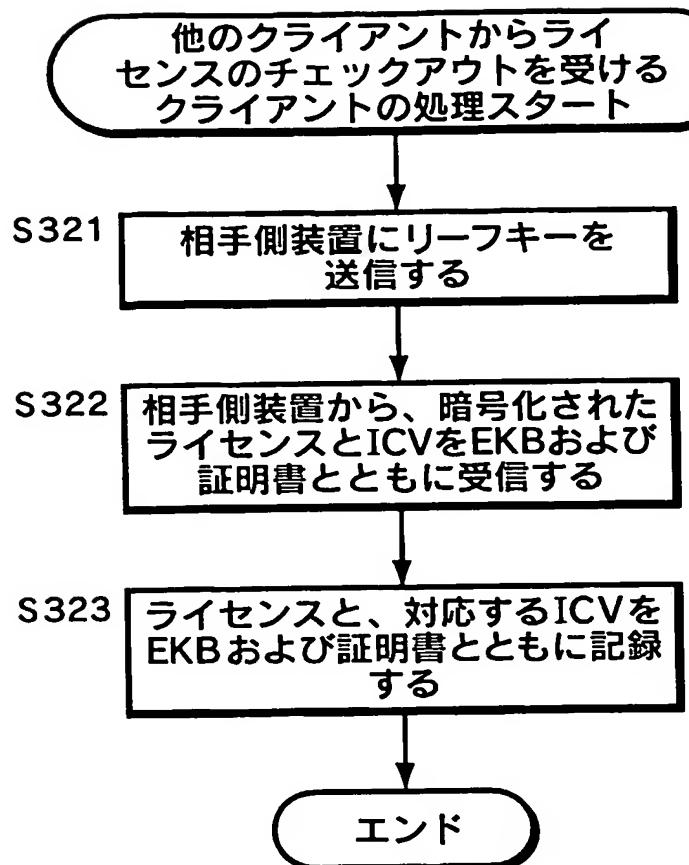


図 44

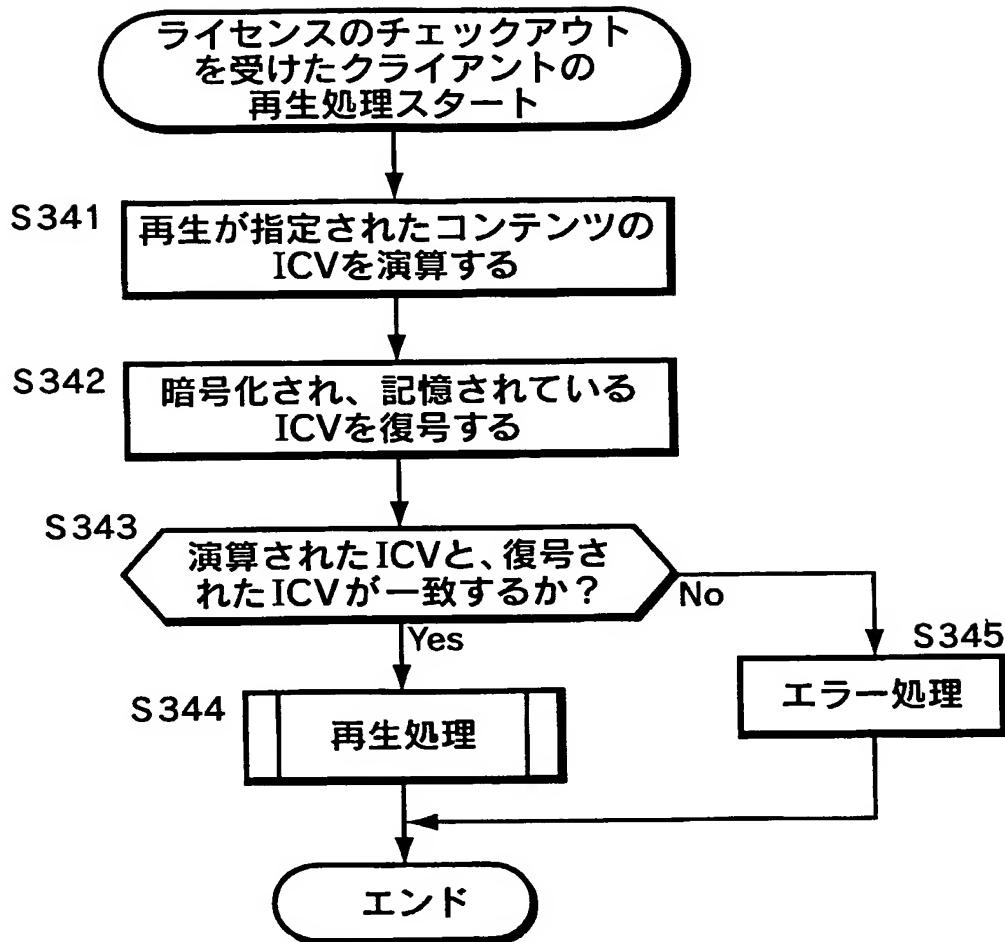
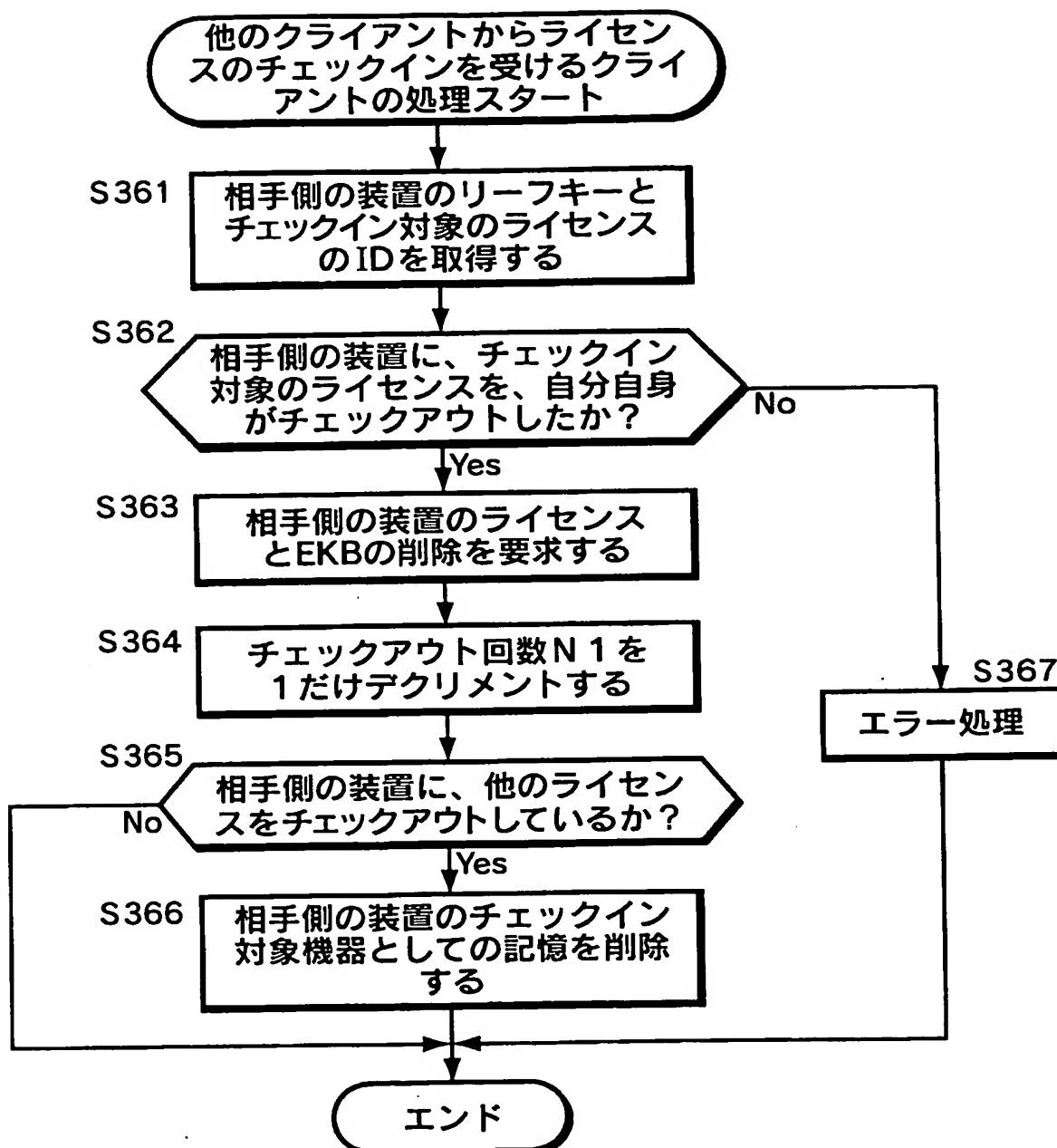


図 45



40/45

図 46

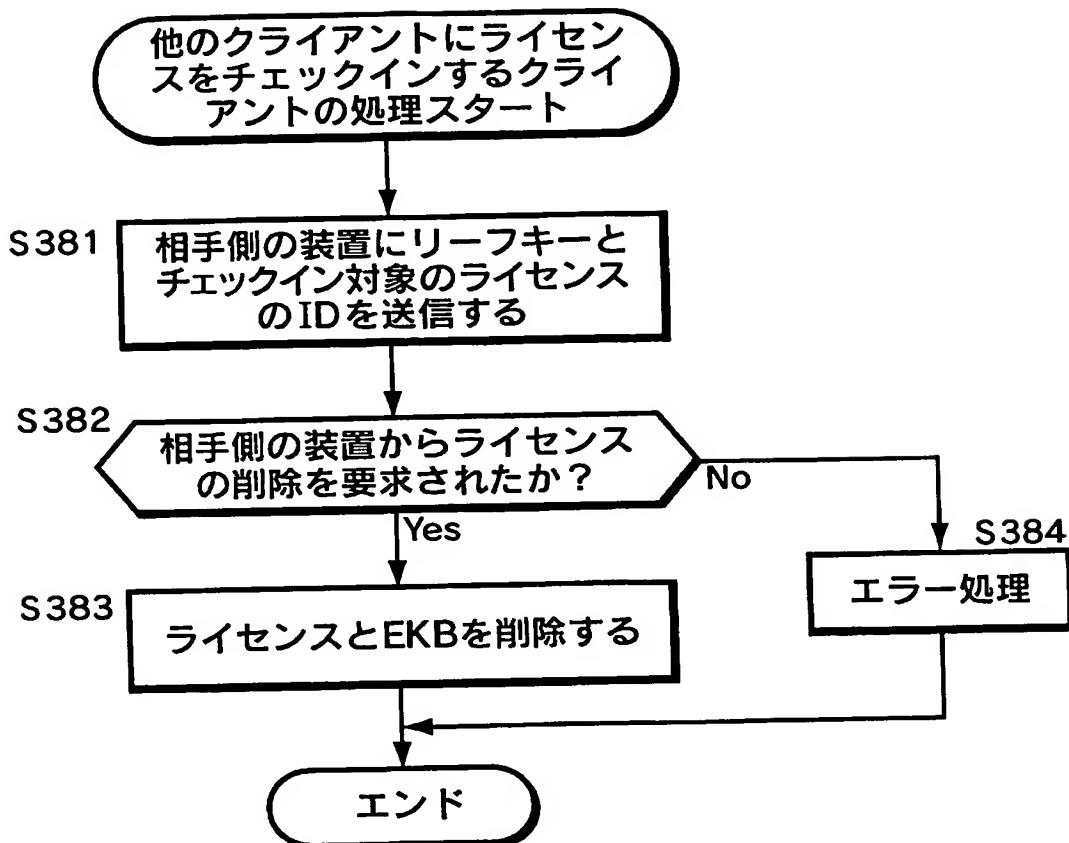


図 47

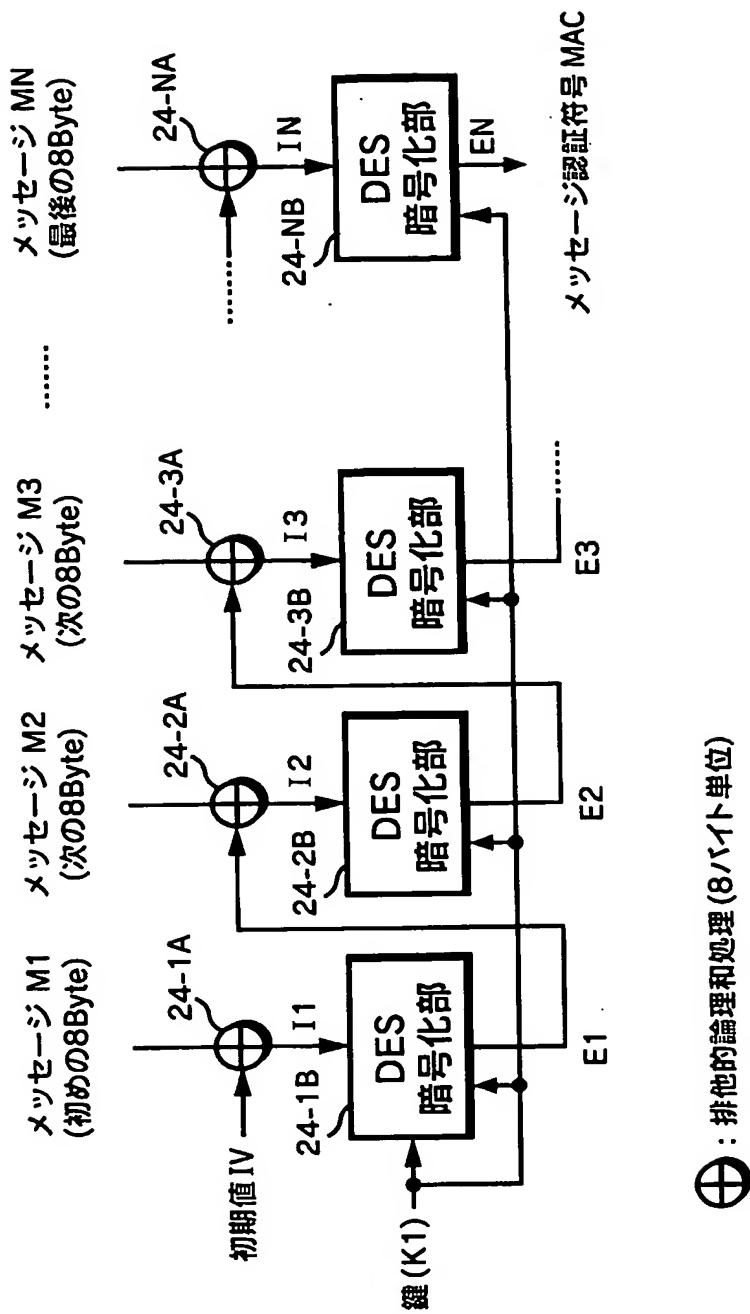
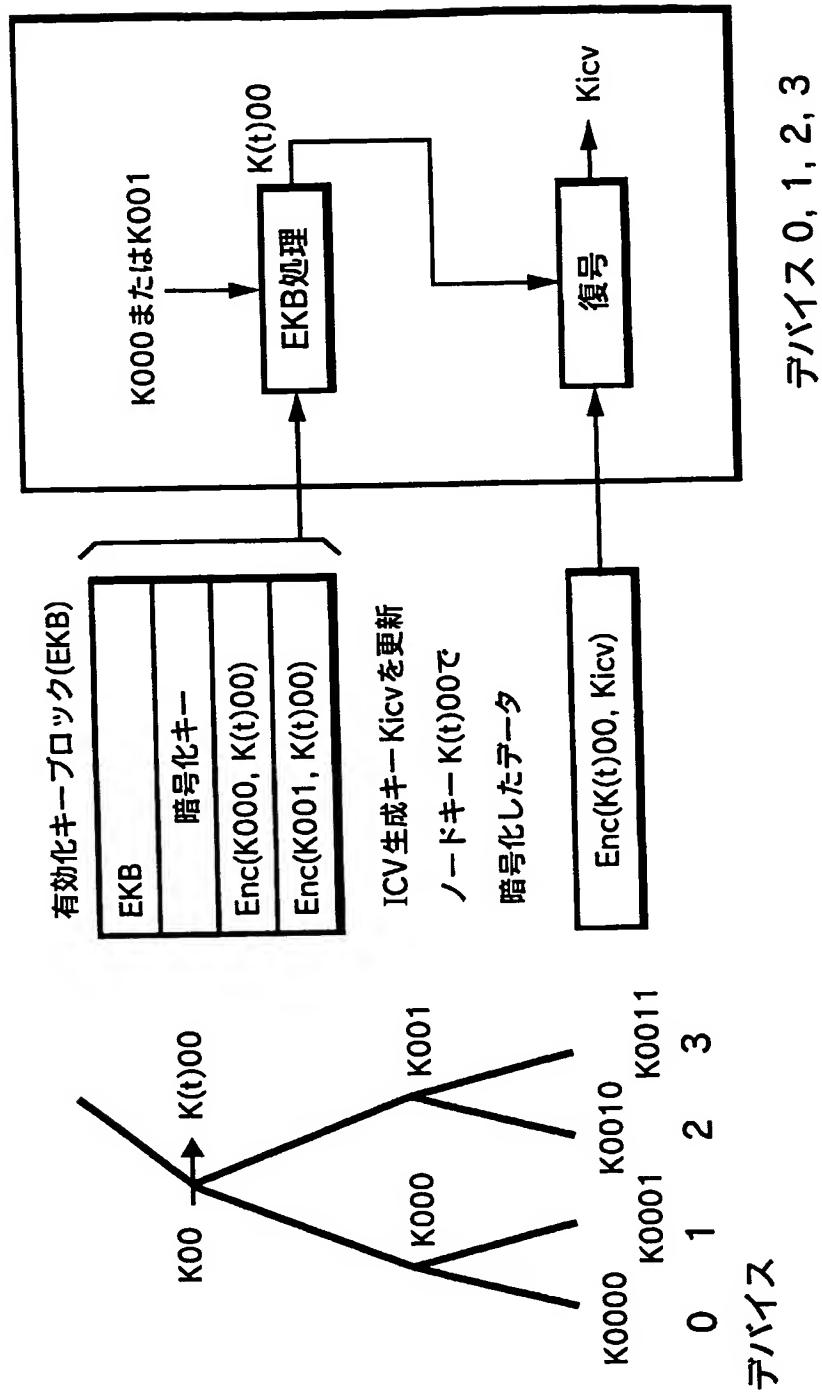
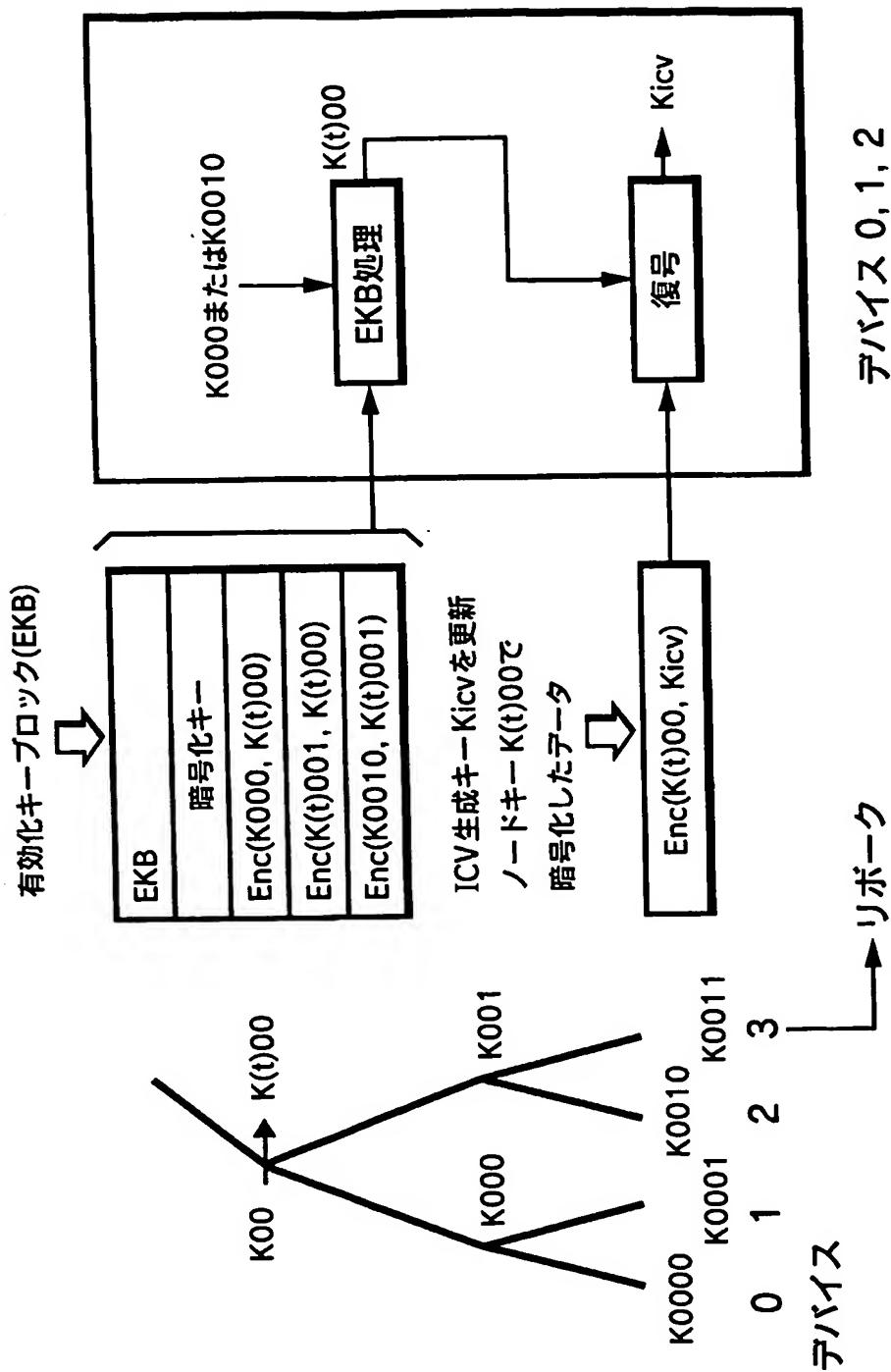
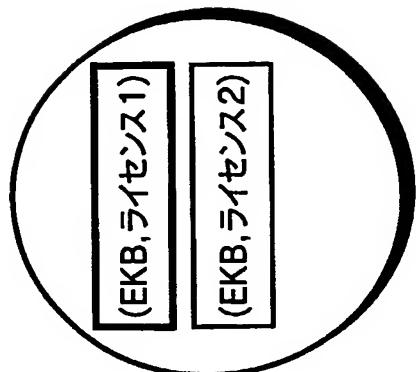


図 48

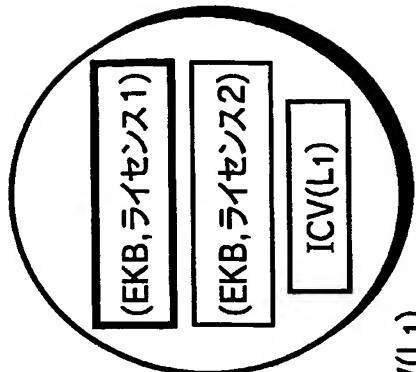


49

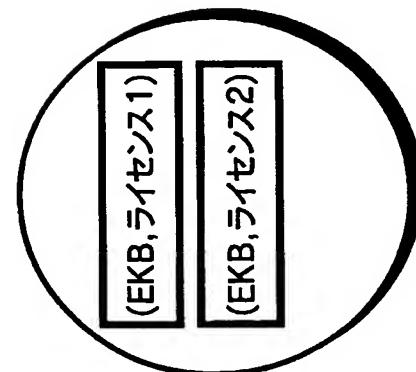




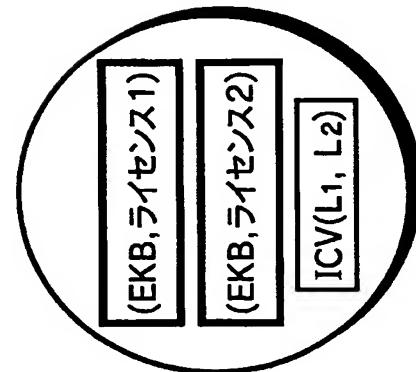
メディア2



メディア2

 $ICV(L_1, L_2) \neq ICV(L_1)$ 

メディア1



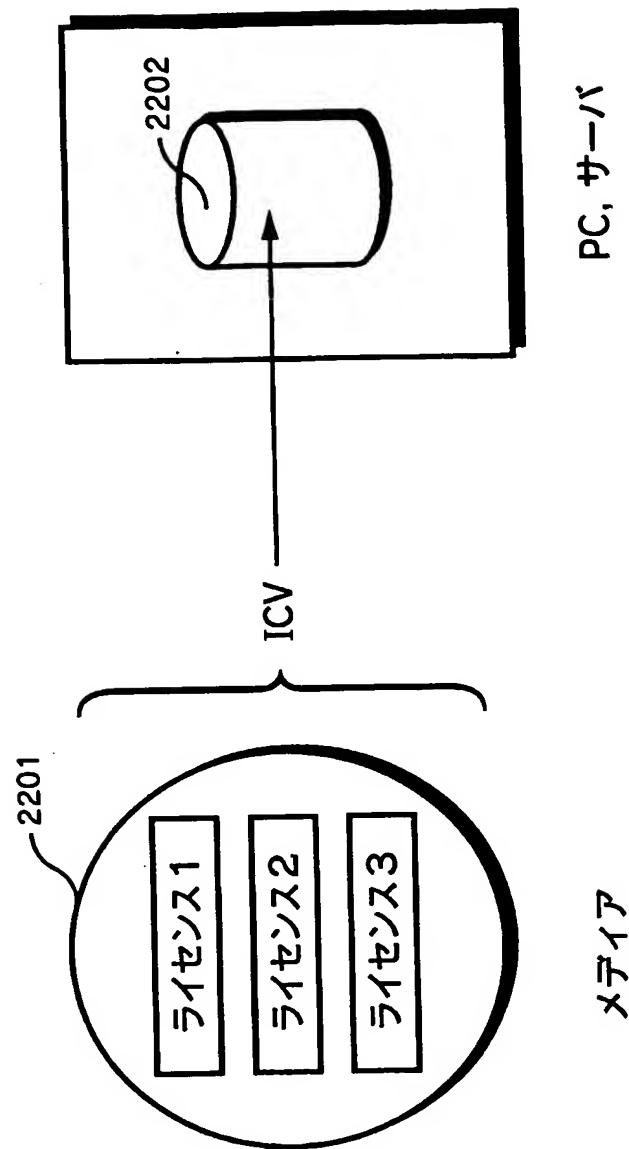
メディア1

図 50A

図 50B

45/45

図51



## INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP02/02955

## A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl<sup>7</sup> H04L9/08, G06F17/60

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl<sup>7</sup> H04L9/08, G06F17/60

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Toroku Jitsuyo Shinan Koho	1994-2002
Kokai Jitsuyo Shinan Koho	1971-2002	Jitsuyo Shinan Toroku Koho	1996-2002

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

JICST FILE (JOIS), WPI, INSPEC (DIALOG)  
 Content, key, tree, revoking, revocation

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WONG, C.K. et al., Secure Group Communications Using Key Graphs. In: Proceedings of ACM SIGCOMM '98, 1988, pages 68 to 79, especially 1.1 Our approach, page 79, left column, lines 25 to 36 < <a href="http://www.acm.org/sigcomm/sigcomm98/tp/technical.html">http://www.acm.org/sigcomm/sigcomm98/tp/technical.html</a> >	1, 4-7, 12-19 2, 3, 8-11
Y	The VersaKey Framework: Versatile Group Key Management, IEEE Journal on Selected Areas in Communications, Vol.17, No.9, pages 1614 to 1631, 1999.09, especially B. Centralized, Tree-Based Key Management	2
Y	US 6049878 A (Sun Microsystems, Inc.), 11 April, 2000 (11.04.00), Column 7, lines 23 to 39 (Family: none)	2

Further documents are listed in the continuation of Box C.  See patent family annex.

* Special categories of cited documents:	
"A" document defining the general state of the art which is not considered to be of particular relevance	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"E" earlier document but published on or after the international filing date	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"O" document referring to an oral disclosure, use, exhibition or other means	"&" document member of the same patent family
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search  
03 July, 2002 (03.07.02)Date of mailing of the international search report  
23 July, 2002 (23.07.02)Name and mailing address of the ISA/  
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

## INTERNATIONAL SEARCH REPORT

International Application No.

PCT/JP02/02955

## C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	Riyo Joken ni Motozuku Anzen na Contents Teikyo Hoshiki, NTT Gijutsu Journal, Vol.12, No.4, pages 26 to 29, 01 April, 2000 (01.04.00), particularly, pages 28 to 29	3, 8-11
A	JP 11-187013 A (IBM Japan, Ltd.), 09 July, 1999 (09.07.99), Par. No. [0017] & CN 1224962 A	1-19

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP02/02955

## Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1.  Claims Nos.:  
because they relate to subject matter not required to be searched by this Authority, namely:
  
2.  Claims Nos.:  
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
  
3.  Claims Nos.:  
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

## Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

The inventions disclosed in the present application are divided into four groups as follows:

1. Claims 1 to 7
2. Claims 8 to 11
3. Claims 12 to 15
4. Claims 16 to 19

1.  As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2.  As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
3.  As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
  
4.  No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

### Remark on Protest

The additional search fees were accompanied by the applicant's protest.  
 No protest accompanied the payment of additional search fees.

## A. 発明の属する分野の分類(国際特許分類(IPC))

Int. Cl' H04L9/08, G06F17/60

## B. 調査を行った分野

調査を行った最小限資料(国際特許分類(IPC))

Int. Cl' H04L9/08, G06F17/60

## 最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	1922-1996年
日本国公開実用新案公報	1971-2002年
日本国登録実用新案公報	1994-2002年
日本国実用新案登録公報	1996-2002年

## 国際調査で使用した電子データベース(データベースの名称、調査に使用した用語)

JICSTファイル (JOIS), WPI, INSPEC (DIALOG)  
content, key, tree, revoking, revocation

## C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
X	WONG, C. K. et al. Secure Group Communications Using Key Graphs. In: Proceedings of ACM SIGCOMM'98, 1998, p. 68-79 especially 1.1 Our approach, p. 79左欄第25-36行 < <a href="http://www.acm.org/sigcomm/sigcomm98/tp/technical.html">http://www.acm.org/sigcomm/sigcomm98/tp/technical.html</a> >	1, 4-7, 12-19
Y		2, 3, 8-11
Y	The VersaKey Framework: Versatile Group Key Management, IEEE Journal on Selected Areas in Communications, Vol. 17, No. 9, p. 1614-1631, 1999. 09 especially B. Centralized, Tree-Based Key Management	2

 C欄の続きにも文献が列挙されている。 パテントファミリーに関する別紙を参照。

## \* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの  
 「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの  
 「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献(理由を付す)  
 「O」口頭による開示、使用、展示等に言及する文献  
 「P」国際出願日前で、かつ優先権の主張の基礎となる出願

## の日の後に公表された文献

「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの  
 「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの  
 「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの  
 「&」同一パテントファミリー文献

## 国際調査を完了した日

03.07.02

## 国際調査報告の発送日

23.07.02

## 国際調査機関の名称及びあて先

日本国特許庁 (ISA/JP)

郵便番号 100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官(権限のある職員)

中里 裕正

5M 9364



電話番号 03-3581-1101 内線 3597

C (続き) . 関連すると認められる文献		関連する 請求の範囲の番号
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	
Y	US 6049878 A (Sun Microsystems, Inc.) 2000.04.11, 第7欄第23-39行 (ワカリなし)	2
Y	利用条件に基づく安全なコンテンツ提供方式, NTT技術ジャーナル, Vol. 12, No. 4, p. 26-29, 2000.04.01, 特にp. 28-29	3, 8-11
A	JP 11-187013 A (日本アイ・ビー・エム株式会社) 1999.07.09, 第17段落 & CN 1224962 A	1-19

## 第I欄 請求の範囲の一部の調査ができないときの意見（第1ページの2の続き）

法第8条第3項（PCT17条(2)(a)）の規定により、この国際調査報告は次の理由により請求の範囲の一部について作成しなかった。

1.  請求の範囲 \_\_\_\_\_ は、この国際調査機関が調査をすることを要しない対象に係るものである。つまり、
2.  請求の範囲 \_\_\_\_\_ は、有意義な国際調査をすることができる程度まで所定の要件を満たしていない国際出願の部分に係るものである。つまり、
3.  請求の範囲 \_\_\_\_\_ は、従属請求の範囲であってPCT規則6.4(a)の第2文及び第3文の規定に従って記載されていない。

## 第II欄 発明の単一性が欠如しているときの意見（第1ページの3の続き）

次に述べるようにこの国際出願に二以上の発明があるとこの国際調査機関は認めた。

この出願の発明は、以下の4群の発明に区分される。

1. 請求の範囲1-7
2. 請求の範囲8-11
3. 請求の範囲12-15
4. 請求の範囲16-19

1.  出願人が必要な追加調査手数料をすべて期間内に納付したので、この国際調査報告は、すべての調査可能な請求の範囲について作成した。
2.  追加調査手数料を要求するまでもなく、すべての調査可能な請求の範囲について調査することができたので、追加調査手数料の納付を求めなかった。
3.  出願人が必要な追加調査手数料を一部のみしか期間内に納付しなかったので、この国際調査報告は、手数料の納付のあった次の請求の範囲のみについて作成した。
4.  出願人が必要な追加調査手数料を期間内に納付しなかったので、この国際調査報告は、請求の範囲の最初に記載されている発明に係る次の請求の範囲について作成した。

## 追加調査手数料の異議の申立てに関する注意

- 追加調査手数料の納付と共に出願人から異議申立てがあった。
- 追加調査手数料の納付と共に出願人から異議申立てがなかった。

THIS PAGE BLANK (USPTO)